

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ  
імені ІГОРЯ СІКОРСЬКОГО»  
Приладобудівний факультет  
Кафедра приладів і систем орієнтації і навігації**

До захисту допущено:

Завідувач кафедри

\_\_\_\_\_ Надія БУРАУ

«\_\_» \_\_\_\_\_ 20\_\_ р.

**Дипломний проєкт  
на здобуття ступеня бакалавра  
за освітньо-професійною програмою «Комп'ютерно - інтегровані технології  
та системи навігації і керування»  
спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології»  
на тему: «Мікромеханічний інклінометр»**

Виконав (-ла):

студент (-ка) IV курсу, групи ПГ-61

Григорій СТРОКАЧ

\_\_\_\_\_

Керівник:

Асистент

Олександр САПЕГІН

\_\_\_\_\_

Рецензент:

Ст. викл., к.т.н

Костянтин БОЖКО

\_\_\_\_\_

Засвідчую, що у цьому дипломному проєкті  
немає запозичень з праць інших авторів без  
відповідних посилань.

Студент (-ка) \_\_\_\_\_

Київ – 2020 року

## Реферат

Пояснювальна записка до дипломного проекту складається з 55 сторінок, містить 28 ілюстрацій, 2 таблиці, 2 додатки, використано 25 джерел.

**Актуальність.** Створення програмного та апаратного середовища Arduino з використання принципу відкритого початкового коду дозволяє у великій кількості створювати свої рішення на основі цих плат. В той же час, введення до програмного середовища Matlab таких пакетів, як Arduino Support Package for Matlab та Simulink Support Package for Arduino Hardware дозволяє ще більш легко працювати з платами Arduino. Ці пакети значно спрощують налаштування та взаємодію з електронними компонентами. Одним з таких компонентів є мікромеханічні датчики. Оскільки мікромеханічні датчики значно менші та дешевші за класичні, це дозволяє використовувати їх у різноманітних приладах та системах де не потрібна висока точність, проте є жорсткі вимоги що до розмірів та енергоживлення. Створення програмного забезпечення для мікромеханічних датчиків та платформи Arduino за допомогою функцій середовища Matlab надає широкі перспективи для синтезу різноманітних приладів і систем орієнтації та навігації.

**Мета проекту.** Метою проекту є розробка апаратного та алгоритмічно-програмного забезпечення для прототипу мікромеханічного інклінометру на основі мікромеханічних чутливих елементів та мікроконтролеру Arduino за допомогою середовища Matlab.

**Перелік ключових слів:** кутомір, мікромеханічний датчик, Arduino, Matlab.

# Зміст

Вступ.....	10
Розділ 1 Сучасні засоби визначення кутового положення .....	11
Поняття про інклінометр.....	11
Безплатформна інерціальна система орієнтації .....	14
Чутливі елементи БІСО .....	15
Платформа Arduino .....	16
Мета і завдання роботи.....	17
Розділ 2 Апаратне та програмне забезпечення інклінометру.....	19
Алгоритм інклінометру .....	19
Кінематика об'єкту .....	19
Алгоритми безплатформної інерціальної системи орієнтації .....	21
Інтегрування .....	23
Схема початкової виставки .....	24
Калібрування чутливого елемента .....	25
Програмне забезпечення .....	31
Виконання алгоритму інклінометру за допомогою пакету Arduino Support Package for Matlab .....	32
Виконання алгоритму інклінометру за допомогою пакету Simulink Support Package for Arduino Hardware .....	34
Конструкція інклінометра .....	38
Компоненти інклінометра .....	38
Конструктивна схема інклінометра.....	40
Корпус інклінометра.....	42
Висновки до розділу 2 .....	44

Розділ 3 Моделювання інклінометру .....	47
Висновки до розділу 3 .....	53
Висновки .....	55
Література .....	57
Додаток А.....	60
Додаток Б .....	64
Додаток В .....	66

## **Перелік скорочень, умовних позначень**

ДКШ – датчика кутової швидкості.

БІСО – безплатформна інерціальна система орієнтації.

ЧЕ – чутливий елемент.

ММГ – мікро-механічний гіроскоп.

ММА – мікро-механічний акселерометр.

МЕМС – мікро-електромеханічна система.

МНК – матриця напрямних косинусів

## Вступ

Розвиток технологій виготовлення мікроелектромеханічних чутливих елементів систем орієнтації дозволяє широко використовувати класичні методи і підходи інерціальної навігації для різних об'єктів. Використання мікромеханіки дозволяє відмовитись від великогабаритних, важких і складних систем у різних сферах. Хоча, мікромеханічні пристрої значно поступаються точністю класичним системам, проте дешевизна та мініатюрність дозволяє використовувати їх у системах низької та середньої точності [1].

Винайдення транзистора співробітниками Bell Laboratories у 1948 році призвело до стрімкого розвитку мікроелектроніки, так що у 1965 році був створений закон Мура, відповідного до якого розміри та щільність розміщення компонентів на комп'ютерному чіпі, а також їх ціна, будуть зменшуватись в двічі кожні два роки. Таке зменшення та розмірів ціни дозволило з'являтися все більшій кількості доступних звичайному користувачеві мікроконтролерів та електронних компонентів. Така тенденція призвела до того, що працювати з електронною технікою може будь-хто за умови наявності відповідних навичок та обладнання. Найкращим прикладом загально доступної електроніки є продукція компанії Arduino. Головною ідеєю Arduino є простота роботи з програмним та апаратним забезпеченням, що дозволяє розширити коло користувачів системи і власноруч створювати різні рішення для практично будь-яких задач [2–4].

Поєднання мікроелектричних чутливих елементів та апаратної платформи Arduino дозволить створити аналоги великогабаритних пристроїв на основі класичних методів і підходів інерціальної навігації при значно менших розмірах ціні та енергозатратах.

## Розділ 1 Сучасні засоби визначення кутового положення

### Поняття про інклінометр

Інклінометр (кутомір, кренометр) – прилад що використовується для визначення куту нахилу об'єкта відносно площини горизонту. Може встановлюватись на автомобілях, кораблях, автокранах, кранових стрілках, землерийних машинах, бурових установках, у високо-поверхових будинках з метою недопущення небезпечних для кренів об'єкта, що можуть призвести до його перекидання або ж відхилення від вертикального положення [5,6,7].

Найпростішим інклінометром є такий, що оснований на роботі сили тяжіння і, по суті, являється фізичним маятником. Вантаж підвішений на нитці, або ж стрілка, що підвішена на підвісі є показчиком і вказує вертикально вниз. Відхилення основи, від вертикального положення, залишає незмінним у просторі положення показчика, проте, зі сторони спостерігача, здається, що показчик відхилився. Кут нахилу реєструється шкалою. Приклад такого інклінометру, що вимірює нахил по кутам крену зображено на рис. 1.1. Перевагою такого інклінометру є простота його конструкції та дешевизна виготовлення. Недоліком - реагування на прискорення об'єкта, що вносить похибку в його покази [8,9].



Рисунок 1.1 – Кренометр з криголаму «Г. Седов»

Іншими можливими чутливими елементами для інклінометру може слугувати рідинний рівень, гіровертикаль, рідинний маятниковий перемикач, акселерометр, триступеневий гіроскоп в кардановому підвісі, датчик кутової швидкості [8,10].

Використання рідинного рівня, рідинного маятникового перемикача, акселерометра має такі ж недоліки, як і фізичний маятник – на їх покази впливає прискорення об'єкту. Наприклад, розглянутий у роботі [11] інклінометр використовує в якості чутливих елементів акселерометри. Але не зважаючи, що він показує досить точні результати нахилу в нерухомому становищі, при його роботі на рухомому об'єкті на чутливі елементи впливають прискорення об'єкту, що вносить в покази приладу значні похибки. Приклад таких похибок зображено на рис. 1.2.

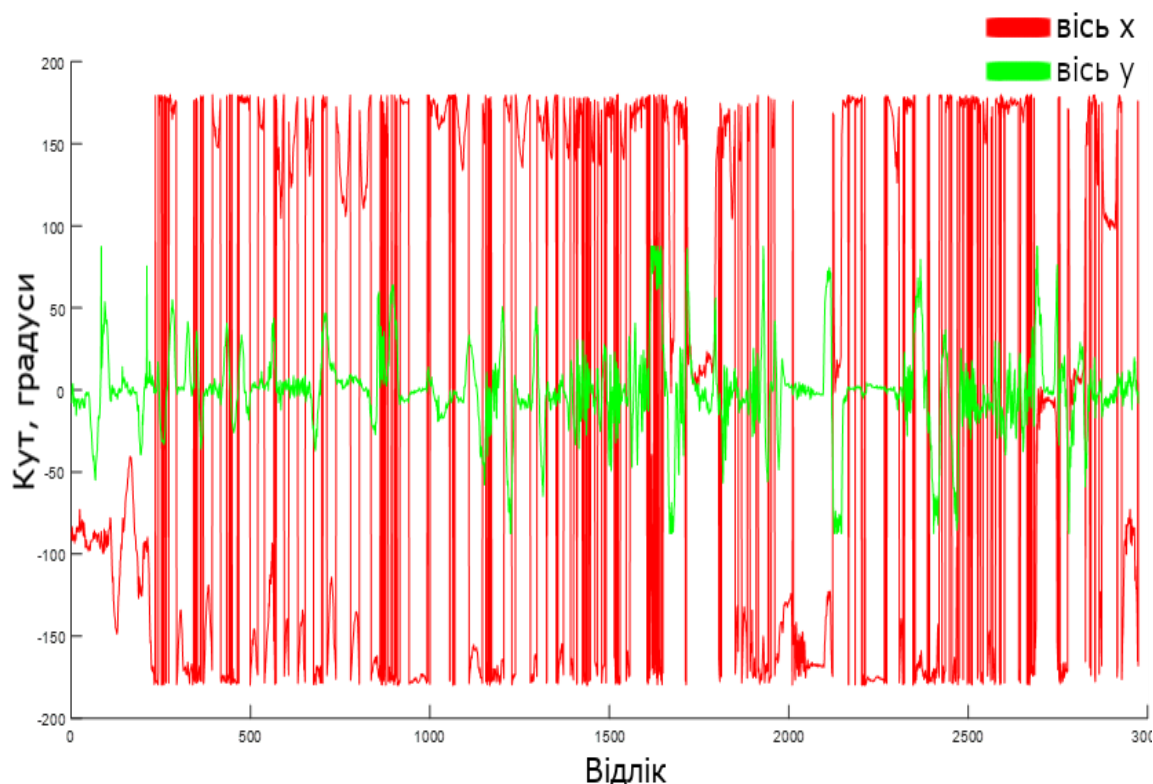


Рисунок 1.2 Покази інклінометру на базі акселерометрів при роботі на рухомому об'єкті

Так, для випадку зображеного на рис. 1.2, вісь «x» трьохосьового блоку акселерометрів була направлена по ходу руху об'єкту, а вісь «y»,



перпендикулярно до нього. Імітування нахилів об'єкту здійснювалося безпосереднім нахилом приладу. І як можна побачити з вище вказаного рис. 1.2 покази по осі «х», яка і була направлена по продольній осі об'єкту, має хаотичний характер, що зовсім не відповідає заданим кутам нахилу приладу.

Гіровертикаль, як прилад, що призначений для орієнтації об'єкта відносно площини горизонту (осі вертикалі), може давати інформацію про нахил об'єкту відносно площини горизонту. Зазвичай гіровертикаллю є триступеневий, астатичний гіроскоп в кардановому підвісі з вертикальною віссю обертання ротора. Також можлива побудова гіровертикалей на основі двохосового гіростабілізатора в індикаторному або силовому виді. В індикаторній побудові – гіроскоп грає роль індикатора відхилення. У силовому гіростабілізаторі гіроскоп безпосередньо бере участь в стабілізації платформи за допомогою силового підвісу. Перевагами такого чутливого елементу приладу є точність. Недоліками є – габарити та вага приладу, необхідність корекції [12].

При побудові інклінометру на основі двоступеневих гіроскопів у вигляді датчиків кутової швидкості (ДКШ) визначення кутів нахилу визначається інтегрування кутової швидкості, яку безпосередньо і вимірюють ДКШ. Проте через помилки інтегрування і, як наслідок, постійне накопичення дрейфу показів, інклінометр на таких датчиках необхідно постійно корегувати. В якості корекції зазвичай використовують показання акселерометрів. А таке об'єднання ДКШ та акселерометрів для зручності об'єднують в безплатформну інерціальну систему орієнтації (БІСО). Проте проблема накопичення похибки все одно має місце дрейф гіроскопів, похибки акселерометрів, не ортогональність вимірювачів та ін. Тому для усунення похибки приладів використовують різні види фільтрів, наприклад, фільтри засновані на алгоритмі Калмана, фільтри Махоні та Маджвіка тощо [12]. Наприклад, інклінометри, що виготовляються компанією Wit-Motion (рис. 1.3), для визначення кутів нахилу приладу використовують мікро-електромеханічні датчики MPU6050 та MPU9250 й працюють на фільтрі Калмана.



Рисунок 1.3 Інклінометри на мікро-електромеханічних датчиках

Перевагами таких інклінометрів є – малі розміри та габарити конструкції. Недоліками – необхідність корекції, низька точність.

### Безплатформна інерціальна система орієнтації

Безплатформною інерціальною системою орієнтації називають таку систему, в якій блок чутливих елементів (ЧЕ) жорстко закріплений на борту об'єкту, що рухається. Тобто осі чутливості ЧЕ змінюють своє положення в просторі разом з об'єктом. Загальними для багатьох БІСО є ЧЕ, що складається з 3-осьового блоку акселерометрів та 3-осьового блоку гіроскопів. Основною задачею БІСО є визначення поточної орієнтації об'єкту відносно певної системи координат [13]. Методика визначення орієнтації залежить від вимірювачів параметрів кутової швидкості. Можливим видом гіроскопічних вимірювачів є такі, що вимірюють [12]:

- кутові швидкості об'єкта у зв'язаній з об'єктом системі координат (лазерний гіроскоп, твердотільний вібраційний гіроскоп, волоконо-оптичні гіроскопи, двоступеневий гіроскоп, мікро-електромеханічний гіроскоп) – датчики кутової швидкості;
- кути повороту об'єкта (одноосьові гіростабілізатори, інтегрований вихідний сигнал ДКШ);

- напрямні косинуси (гіроскопи зі сферичним ротором – електростатичні, магнітні, з газовим підвісом).

Кінцевою метою роботи БІСО є отримання трьох кутів, що характеризують відхилення від початкової системи координат – курс, крен, тангаж.

Перевагами БІСО є [12]:

- перешкодостійкість;
- висока швидкість визначення та передачі даних;
- незбурюваність зовнішніми прискореннями, а як наслідок, відсутність перешкод по всім вихідними параметрам.

Недоліками БІСО є:

- необхідність задання початкового положення;
- необхідність безперервності роботи;
- збільшення похибок у часі.

### **Чутливі елементи БІСО**

Гіроскопи, що забезпечать високу точність мають досить велику розміри та вагу, що є проблемою для багатьох портативних рішень. Мікро-механічні гіроскопи (ММГ) та мікро-механічні акселерометри (ММА), що входять до складу мікро-електромеханічних систем (МЕМС), хоч і програють у точності своїм більшим побратимам, мають значно менші розміри й, відповідно, низьке електроспоживання. Це дозволяє інтегрувати МЕМС у різні види пристроїв – смартфони, дрони, ігрові аксесуари, тощо. МЕМС датчики досить вигідно об'єднувати в один корпус. Так більшість рішень об'єднують на одному чіпі в блок інерціальних вимірювань, який складається з 3-осьового блока ММГ та 3-осьового блока ММА, що вже достатньо для побудови базової системи орієнтації. В деяких випадках до гіроскопів й акселерометрів додають магнітометри [14,15,16,17].

На ринку МЕМС датчиків на сьогоднішній день представлено багато різних рішень з різними характеристиками. В таких МЕМС рішеннях на одній платі об'єднано трьох-осьовий блок датчиків кутових швидкосте та трьох-осьовий блок

акселерометрів, що є необхідним для вирішення задачі. Характеристики деяких загальнодоступних датчиків представлені в таблиці 1.1

Таблиця 1.1 Характеристики деяких МЕМС датчиків

Найменування датчика	MPU6050	MPU9250	lsm9ds1	BNO055
Напруга живлення	3,5 В; 5 В	2.4В; 3,6В	1,9В; 3,6В	2.4В; 3,6В
Діапазон вимірювань акселерометру, g	$\pm 2$ , $\pm 4$ , $\pm 8$ та $\pm 16$ g	$\pm 2$ , $\pm 4$ , $\pm 8$ та $\pm 16$ g	$\pm 2$ , $\pm 4$ , $\pm 8$ та $\pm 16$ g	$\pm 2$ , $\pm 4$ , $\pm 8$ та $\pm 16$ g
Діапазон вимірювань гіроскопу, град/с	$\pm 250$ , $\pm 500$ , $\pm 1000$ та $\pm 2000$ °/с	$\pm 250$ , $\pm 500$ , $\pm 1000$ та $\pm 2000$ °/с	$\pm 245$ , $\pm 500$ та $\pm 2000$ °/с	$\pm 125$ до $\pm 2000$ °/с
Шум гіроскопа, °/с/ $\sqrt{\text{Гц}}$	0,005	0,01	-	0,014
Інтерфейс підключення	I2C	I2C, SPI	I2C, SPI	I2C, UART

### Платформа Arduino

Здешевлення виготовлення електронних компонентів дало змогу збирати на їх основі різні рішення. Розвиток і здешевлення сучасної електроніки дозволило створити відкритий апаратно-програмний продукт Arduino. Arduino пропонує своїм користувачам легкий спосіб для програмування мікроконтролерів і створення на основі готових плат пристроїв. Відкритість архітектури дозволяє користувачам створювати власні модифікації та варіації для плати. На сьогоднішній день найбільш розповсюджені типи наведено в таблиці 1.2 [4].

Таблиця 1.2 Деякі характеристики плат Arduino

Плата Arduino	Мікроконтролер	Вихідна напруга, В	Вихідний струм, мА	Кількість цифрових пінів	Falsh пам'ять, кБ
Uno Rev3	ATmega328P	5	20	14	32
Nano	ATmega328P	5	20	22	32
Leonardo	ATmega32u4	5	50	20	32
Mega 2560	ATmega2560	5	50	54	256

Rev3					
------	--	--	--	--	--

Програмування плат відбувається через офіційну програму «Arduino IDE». Проте можливі й інші шляхи, так в програмному середовищі Matlab існує два пакети «Arduino Support Package for Matlab» та «Simulink Support Package for Arduino Hardware». Ці два пакети дозволяють розробляти алгоритм для плат та працювати з ним в режимі реального часу, а у випадку з пакетом для Simulink можливе вивантаження створеної програми на плату.

Загальними для обох пакетів є курування такими стандартними функціями як:

- налаштування цифрових пінів на вхід або вихід
- зчитування з аналогового входу
- генерування сигналу для широтно-імпульсного модулятора
- робота з пристроями на основі інтерфейсу «I2C»
- робота з пристроями на основі інтерфейсу «SPI»
- робота з пристроями на основі інтерфейсу «Serial»
- робота з серво-машинами

В останніх версіях пакетів додалися окремі функції для роботи з найбільш популярними MEMS сенсорами – MPU6050, MPU9250, LSM9DS1, BNO055. Ці функції дозволяють легко налаштовувати підключення з датчиками та, за умови наявності тулбоксу «Sensor Fusion and Tracking Toolbox™ and Navigation Toolbox™» задавати для них: частоту опитування датчиків, кількість відліків при зчитуванні й вихідний формат даних у Matlab; діапазон і частота зчитування показань гіроскопів, діапазон і частота зчитування показань акселерометрів, зчитування температури, зчитування показань магнітометрів (для датчиків з магнітометрами) у Simulink [18, 19].

### **Мета і завдання роботи**

Метою проекту є розробка апаратного та алгоритмічно-програмного забезпечення для прототипу мікромеханічного інклінометру на основі

мікромеханічних чутливих елементів та мікроконтролеру Arduino за допомогою середовища Matlab.

Завданням є розробка працюючого прототипу інклінометра, який повинен працювати на основі плат Arduino. Головним чутливим елементом приладу виступатиме МЕМС інерціальний вимірювальний модуль (ІВМ). Алгоритм розроблюється у програмному середовищі Matlab через який, в том числі, виконується вивантаження програмного забезпечення на плату.

Розроблюваний алгоритм має мати достатню точність та швидкодію для відображення достовірної інформації про кути нахилу об'єкту, на якому встановлено прилад. Також алгоритм має протидіяти зовнішнім хибним збуренням, що можуть внести похибку у покази приладу.

Прилад повинен забезпечувати комфортність користування і працювати в автономному режимі. Для цього необхідно розробити корпус приладу, який буде дозволяти встановлювати прилад на широкий спектр поверхонь та давати зручну для зчитування інформацію про кути нахилу приладу, а відповідно і об'єкту на якому встановлено прилад.

## Розділ 2 Апаратне та програмне забезпечення інклінометру

### Алгоритм інклінометру

#### Кінематика об'єкту

Інклінометр має показувати поточну орієнтацію об'єкта в просторі. Введемо зв'язану з вимірювальними осями ІВМ систему координат  $1_p 2_p 3_p$ . У якості опорної СК використаємо географічну  $1_s 2_s 3_s$ . Зв'язана СК нахилена відносно опорної на кути рискання  $\psi$ , крену  $\theta$  і тангажу  $\varphi$ , як показано на рис.2.1 [20].

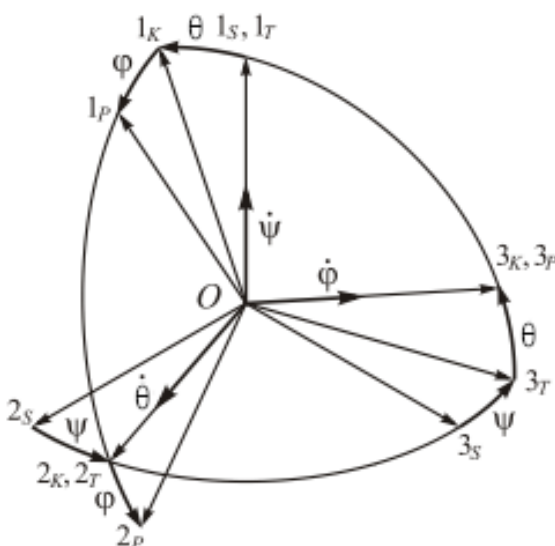


Рисунок 2.1 Кути Ейлера-Крилова

Орієнтація зв'язаної СК відносно опорної визначається, за рис.2.1, відповідною матрицею напрямних косинусів (МНК)  $C^{SP}$ :

$$\begin{array}{c|ccc}
 C^{SP} & 1_p & 2_p & 3_p \\
 \hline
 1_s & \cos \theta \cos \varphi & -\cos \theta \sin \varphi & \sin \theta \\
 2_s & \cos \psi \sin \varphi + \sin \psi \cos \varphi \sin \theta & \cos \psi \cos \varphi - \sin \psi \sin \varphi \sin \theta & -\sin \psi \cos \theta \\
 3_s & \sin \psi \sin \varphi - \cos \psi \cos \varphi \sin \theta & \sin \psi \cos \varphi + \cos \psi \sin \varphi \sin \theta & \cos \psi \cos \theta
 \end{array} \quad (2.1)$$

Відомим кінематичним параметром також є кватерніон повороту, а саме параметри Родріга-Гамільтона:

$$Q = q_0 + iq_1 + jq_2 + kq_3 \quad (2.2)$$

У кожного способу подання орієнтації є свої переваги та недоліки. Для кутів Ейлера-Крилова перевагами виступатиме наочність подання поворотів, а недоліками – відсутність операції складання поворотів, непристосованість до безпосередньої проекції векторів. У випадку МНК перевагами є пристосованість до проекції векторів та існування простого правила додавання поворотів, недоліки – не наочність здійснених поворотів, велика кількість параметрів опису повороту. Для кватерніона повороту перевагами будуть: мала кількість параметрів, просте правило складання поворотів, пристосованість до перепроєціювання векторів; недоліками стануть відсутність наочності подання повороту [13].

Рівнянням орієнтації називають рівняння, розв'язування яких дає змогу визначити відповідні параметри повороту, які характеризуватимуть орієнтацію тіла відносно певної системи координат [12].

Кінематичне рівняння Ейлера використовує кути Ейлера-Крилова й можна подати у вигляді [13]:

$$\begin{cases} \frac{d\varphi}{dt} = (\omega_{1P}^{PS} \cdot \cos \varphi - \omega_{2P}^{PS} \cdot \sin \varphi) / \cos \varphi; \\ \frac{d\theta}{dt} = \omega_{2P}^{PS} \cdot \cos \varphi + \omega_{1P}^{PS} \cdot \sin \varphi; \\ \frac{d\psi}{dt} = \omega_{3P}^{PS} - \operatorname{tg} \theta (\omega_{1P}^{PS} \cdot \cos \varphi - \omega_{2P}^{PS} \cdot \sin \varphi). \end{cases} \quad (2.3)$$

Недоліком цього рівняння буде їх некоректність при куті  $\theta = \pm 90^\circ$ , через що виникає необхідність штучного обмеження кутів тіла, або алгоритмічно [12].

Кінематичне рівняння Пуассона використовує МНК і має форму [13]:

$$\frac{dC^{SP}}{dt} = C^{SP} \cdot (\omega_P^{PS} \times). \quad (2.4)$$

Тут використано позначення кососиметричної матриці

$$(\omega_P^{PS} \times) = \begin{bmatrix} 0 & -\omega_{3P}^{PS} & \omega_{2P}^{PS} \\ \omega_{3P}^{PS} & 0 & -\omega_{1P}^{PS} \\ -\omega_{2P}^{PS} & \omega_{1P}^{PS} & 0 \end{bmatrix}. \quad (2.5)$$



Матриця  $(\omega_p^{PS} \times)$  формується з сигналів гіроскопів IBM.

Кватерніонне рівняння орієнтації виглядає наступним чином

$$\frac{dQ^{SP}}{dt} = \frac{1}{2} Q^{SP} \circ \Omega_P^{PS}, \quad (2.6)$$

де  $Q^{SP}$  - кватерніон повороту від початкової системи координат S до поточної P,  $\Omega_P^{PS}$  - вектор-кватерніон, скалярна частина якого є нулем, а векторна складається з проекцій  $\omega_p^{PS}$  на осі системи P вектора  $\omega^{PS}$  кутової швидкості повороту системи координат P по відношенню до системи координат S,  $\circ$  - позначення кватерніонного добутку [13].

### **Алгоритми безплатформеної інерціальної системи орієнтації**

Найпростішим алгоритмом для створення інклінометру стане визначення кутового положення тіла за допомогою акселерометрів відносно вектору вільного падіння. Проте, як було зазначено у першому розділі, алгоритм працюючий лише на акселерометрах має суттєвий недолік – хибність показів при зовнішніх збуреннях, що діють на акселерометри.

Іншим способом, для отримання кутового положення тіла відносно певної початкової системи координат, є виконання перетворення сигналів датчиків кутової швидкості та прискорення. Для цього датчики встановлюють на спеціальну гіростабілізовану платформу, яка утримує акселерометри в заданій системі координат, або використовують блок гіроскопів та акселерометрів з вимірювачем, які жорстко встановлюють на об'єкті, кутове положення якого вимірюють. Розглянемо випадок, коли датчики жорстко встановлені на об'єкті [12].

Отже мета алгоритму, для жорстко на об'єкті закріплених датчиків, полягає у перетворенні даних про переміщення об'єкта в його поточне кутове положення відносно певної системи координат. Виберемо в якості запису кутового положення об'єкту МНК. Початкова МНК може визначатись за допомогою

початкової виставки або прийнятись як матриця, що повністю збігається з площиною горизонту.

З кутової швидкості формується кососиметрична матриця, яка, разом з МНК минулого кроку (або початкової виставки) використовуються у кінематичному рівнянні, наприклад Пуассона. Отриманий вираз з рівняння Пуассона інтегрується чисельним методом. Результатом інтегрування є МНК, що відображає нове положення тіла.

На даному етапі вже можливо розрахувати з МНК три кута повороту, проте через накопичення похибки інтегрування, така МНК не буде відображати реального положення тіла. Через що виникає потреба у компенсації похибки. Для цього можна ввести корекцію за допомогою фільтрів. Такими фільтрами можуть виступати фільтри, що базуються на методі Калмана, комплементарний фільтр, фільтри Махоні, Маджвіка, тощо.

Найпростішим з перерахованих фільтрів є комплементарний фільтр. Його принцип роботи полягає у поєднанні вимірювань гіроскопів з вимірюваннями акселерометрів результатом чого становиться компенсація похибку інтегрування кутової швидкості гіроскопів та неточності показань з акселерометрів у результаті зовнішніх збурень. Так опісля чисельного інтегрування дані з гіроскопів пропускаються через фільтр високих частот, а показання акселерометрів проходять через фільтр низьких частот. Схема роботи комплементарного фільтру приведена на рис. 2.1 [21, 22].

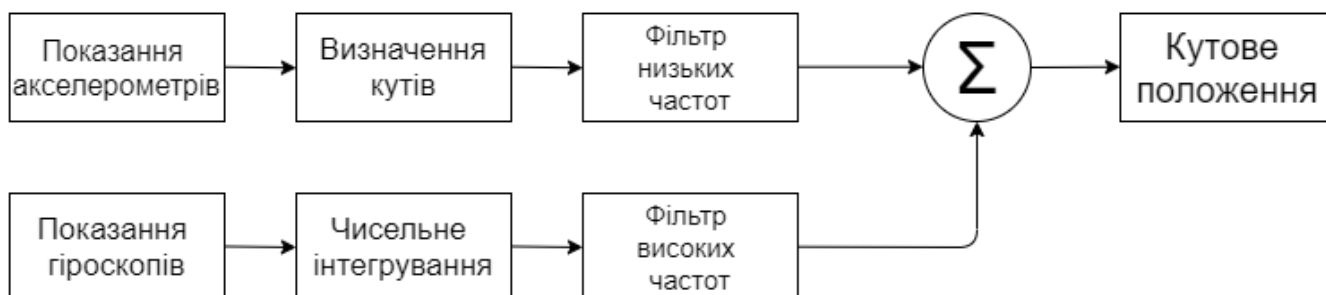


Рисунок 2.2 Схема роботи комплементарного фільтру

Для випадку з МНК з показань акселерометрів вираховуються кути відносно вектору вільного падіння  $\mathbf{g}$  з яких формуються МНК. Ця МНК вноситься у рівняння комплементарного фільтру. Зважаючи на це рівняння фільтру прийме вигляд

$$C^{SP} = (1 - k) \cdot C_{\omega}^{SP} + k \cdot C_a^{SP} \quad (2.7)$$

де  $k$  це коефіцієнт фільтру,  $C_{\omega}^{SP}$  - МНК з показань гіроскопу,  $C_a^{SP}$  - МНК з показань акселерометру,  $C^{SP}$  - вихідна МНК. Величина коефіцієнту фільтрів впливає на вихідні показання фільтру. Так коефіцієнт близький до 1 буде сильно збільшувати долю впливу акселерометрів, що буде повертати проблему з реакцією на прискорення та вібрації. Занадто малий коефіцієнт буде допускати накопичення похибки інтегрування [22].

### Інтегрування

Для визначення кутів із кутової швидкості необхідно виконати чисельне інтегрування. Серед усіх методів чисельного інтегрування найбільш простими є такі методи:

- метод прямокутників (лівих, правих, середніх);
- метод трапеції;
- метод парабол (Сімпсона);

Метод прямокутників є найпростішим методом у якому на кожному кроці інтегрування функція змінюється на сталу величину, а для визначення повного інтегралу необхідно просумувати усі задані значення функції, окрім останнього і результат домножити на крок інтегрування. Формула цього методу для відрізка  $[a, b]$  має вигляд:

для лівих прямокутників

$$\int_a^b x(t) dt \approx x(a)(b - a); \quad (2.8)$$

для правих прямокутників

$$\int_a^b x(t)dt \approx x(b)(b-a). \quad (2.9)$$

При використанні методу трапецій обчислюється площа трапецій, що утворюються при лінійній інтерполяції точок функції. Формула методу для відрізка  $[a,b]$  має вигляд:

$$\int_a^b x(t)dt \approx h\left(\frac{x_1 + x_n}{2} + \sum_{i=1}^{n-1} x_i\right) - \frac{h^2(b-a)}{12} X''(\xi). \quad (2.10)$$

Для методу Сімпсона виконується поділ інтервалу інтегрування  $[a,b]$  на парну кількість частин так, щоб загальна кількість точок стала непарною, а крок інтегрування дорівнювати:

$$h = \frac{b-a}{n-1} = \frac{b-a}{2(k-1)}. \quad (2.11)$$

Для кожного інтервалу виконується інтерполяція заданих точок квадратною параболою інтегруючи яку в інтервалі, отримують формулу Сімпсона для окремої ділянки. Підсумовуючи результат по усіх частинних відрізках, отримують квадратурну формулу Сімпсона:

$$\int_a^b x(t)dt \approx \frac{h}{3}(x_1 + 4x_2 + 2x_3 + 4x_4 + \dots + 4x_{n-1} + x_n) - \frac{h^4(b-a)}{180} X^{IV}(\xi). \quad (2.12)$$

Для зменшення похибки будь-якого методу необхідно збільшувати кількість заданих точок початкової функції у заданому інтервалі. Ступень зменшення залежить від методу. Показник ступеня похибки залишкових членів інтегрування від кроку прийнято називати порядком метода інтегрування. Так метод прямокутників є методом 1 порядку, метод трапецій – 2, метод Сімпсона – 4 [23].

### **Схема початкової виставки**

Диференціальні рівняння, що використовуються у БІСО, для своєї роботи потребують початкових умов, які задаються в процесі початкової виставки. Такими початковими умовами виступають початкові умови для диференціальних

рівнянь Ейлера, початкова МНК для рівняння Пуассона, початковий кватерніон для рівнянь в кватерніонах [12].

Для визначення кутів будемо вважати, що система в момент визначення початкового положення не рухома. В такому випадку визначити початкові кути буде найпростіше за допомогою показань акселерометрів відносно вектору прискорення вільного падіння. Для МНК (2.1), проекції вектору прискорення сили тяжіння на осі акселерометрів можна визначити за допомогою рівнянь:

$$\begin{aligned} g_x &= g \cos \theta \sin \varphi = g c_{11} \\ g_x &= -g \sin \theta = g c_{13} \\ g_z &= -g \cos \theta \cos \varphi = g c_{12} \end{aligned} \quad , \quad (2.13)$$

де  $c_{11}, c_{12}, c_{13}$  визначається як:

$$c_{11} = \frac{g_x}{g}; \quad c_{12} = \frac{g_z}{g}; \quad c_{13} = -\frac{g_y}{g}. \quad (2.14)$$

Зважаючи на те, що для даної реалізації інклінометру нам не важливо знати кут  $\psi$  (курс), то цілком достатньо визначити кути  $\varphi$  (тангаж) та  $\theta$  (крен). Необхідні кути вираховується за формулами:

$$\begin{aligned} \varphi &= \operatorname{artg}\left(\frac{c_{11}}{c_{12}}\right); \\ \theta &= \operatorname{arctg}\left(\frac{c_{13}}{\sqrt{1-c_{13}^2}}\right). \end{aligned} \quad (2.15)$$

Кут  $\psi$  можна прийняти рівним 0.

З обчислених кутів  $\varphi$ ,  $\theta$  та прийнятого  $\psi$  визначається початкова МНК або інші початкові параметри для диференціальних рівнянь орієнтації.

### Калібрування чутливого елемента

Простим методом калібрування гіроскопа та акселерометра виконується методами тестових поворотів [12].

Калібрування акселерометра викується в діапазоні вимірів  $\pm 1g$  на оптичній ділильній головці (ОДГ). Акселерометр встановлюється однією зі своїх баз до бокової бази базової площадки ОДГ, що забезпечує паралельність осей

акселерометра до осей ОДГ. Так блок акселерометрів встановлений так, щоб вісь чутливості  $x$  була паралельна до осі обертання базової площадки ОДГ і направлена від неї, вісь  $y$  – праворуч від осі  $x$  та вісь  $z$  – вертикально вгору. Надалі, виконуючи обороти навколо осі  $x$  по осі  $y$  та  $z$  задається відоме значення прискорення і в даному випадку це вектор  $\mathbf{g}$ . Однак в цьому випадку не відбувається калібрування осі  $x$ , тому необхідно змінити вісь обертання, наприклад на вісь  $y$ , та виконати повороти вже навколо неї також задаючи відоме значення прискорення  $\mathbf{g}$ .

Таким чином, обертаючи навколо осі чутливості блоку акселерометрів  $x$  виміряно 4 випадки дії прискорення вільного прискорення на осі чутливості блоку акселерометрів:

$$\begin{aligned} a_x &= 0 & a_y &= 0 & a_z &= g \\ a_x &= 0 & a_y &= g & a_z &= 0 \\ a_x &= 0 & a_y &= 0 & a_z &= -g \\ a_x &= 0 & a_y &= -g & a_z &= 0 \end{aligned} \quad (2.16)$$

та 4 випадки для обертання навколо осі чутливості блоку акселерометрів  $y$ :

$$\begin{aligned} a_x &= 0 & a_y &= 0 & a_z &= g \\ a_x &= g & a_y &= 0 & a_z &= 0 \\ a_x &= 0 & a_y &= 0 & a_z &= -g \\ a_x &= -g & a_y &= 0 & a_z &= 0 \end{aligned} \quad (2.17)$$

Для яких вихідне значення акселерометрів буде складати при обертанні навколо осі  $x$ :

$$\begin{aligned} U_{ax1} &= k_{13}g + U_{ax0} & U_{ay1} &= k_{23}g + U_{ay0} & U_{az1} &= k_{33}g + U_{az0} \\ U_{ax2} &= k_{12}g + U_{ax0} & U_{ay2} &= k_{22}g + U_{ay0} & U_{az2} &= k_{32}g + U_{az0} \\ U_{ax3} &= -k_{13}g + U_{ax0} & U_{ay3} &= -k_{23}g + U_{ay0} & U_{az3} &= -k_{33}g + U_{az0} \\ U_{ax4} &= -k_{12}g + U_{ax0} & U_{ay4} &= -k_{22}g + U_{ay0} & U_{az4} &= -k_{32}g + U_{az0} \end{aligned} \quad (2.18)$$

та при обертанні навколо осі  $y$ :

$$\begin{aligned} U_{ax5} &= k_{13}g + U_{ax0} & U_{ay5} &= k_{23}g + U_{ay0} & U_{az5} &= k_{33}g + U_{az0} \\ U_{ax6} &= -k_{11}g + U_{ax0} & U_{ay6} &= -k_{21}g + U_{ay0} & U_{az6} &= -k_{31}g + U_{az0} \\ U_{ax7} &= -k_{13}g + U_{ax0} & U_{ay7} &= -k_{23}g + U_{ay0} & U_{az7} &= -k_{33}g + U_{az0} \\ U_{ax8} &= k_{11}g + U_{ax0} & U_{ay8} &= k_{21}g + U_{ay0} & U_{az8} &= k_{31}g + U_{az0} \end{aligned} \quad (2.19)$$

Для кожного положення замір значень виконувався близько 60 с, що дозволяє майже повністю прибрати вплив шумів вимір провівши осереднення вимірів.

Масштабні коефіцієнти та зміщення нуля розраховуються за формулами:

$$\begin{aligned} k_{11} &= (U_{ax8} - U_{ax6}) / 2g & k_{12} &= (U_{ax2} - U_{ax4}) / 2g & k_{13} &= (U_{ax1} - U_{ax3} + U_{ax5} - U_{ax7}) / 4g \\ k_{21} &= (U_{ay8} - U_{ay6}) / 2g & k_{22} &= (U_{ay2} - U_{ay4}) / 2g & k_{23} &= (U_{ay1} - U_{ay3} + U_{ay5} - U_{ay7}) / 4g \\ k_{31} &= (U_{az8} - U_{az6}) / 2g & k_{32} &= (U_{az2} - U_{az4}) / 2g & k_{33} &= (U_{az1} - U_{az3} + U_{az5} - U_{az7}) / 4g \end{aligned} \quad (2.20)$$

$$U_{ax0} = \frac{1}{8} \sum_{i=1}^8 U_{axi}; \quad U_{ay0} = \frac{1}{8} \sum_{i=1}^8 U_{ayi}; \quad U_{az0} = \frac{1}{8} \sum_{i=1}^8 U_{azi}; \quad (2.21)$$

Розраховані коефіцієнти формуються у матриці:

$$K = \begin{bmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{bmatrix}, \quad (2.22)$$

$$U_{a0} = \begin{bmatrix} U_{ax0} \\ U_{ay0} \\ U_{az0} \end{bmatrix}. \quad (2.23)$$

Калібрування гіроскопу відбувається в два етапи – задання тестових значень кутової швидкості та прискорень. В першому задається декілька значень кутових швидкостей навколо кожної осі чутливості. На другому етапі інерціально-вимірювальний модуль встановлюється в різних положеннях відносно вектору прискорення сили тяжіння  $g$ , так як і при калібруванні акселерометрів, що описано вище.

Модель вихідного сигналу блоку гіроскопів має вигляд:

$$\begin{bmatrix} U_{\omega x} \\ U_{\omega y} \\ U_{\omega z} \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \cdot \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} + \begin{bmatrix} U_{\omega x0} \\ U_{\omega y0} \\ U_{\omega z0} \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix} \cdot \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} + \begin{bmatrix} n_{\omega x} \\ n_{\omega y} \\ n_{\omega z} \end{bmatrix}. \quad (2.24)$$

Але, зважаючи на те, що датчик має не дуже високу точність можемо знехтувати впливом прискорення сили тяжіння на гіроскоп.

$$\begin{bmatrix} U_{\omega x} \\ U_{\omega y} \\ U_{\omega z} \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \cdot \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} + \begin{bmatrix} U_{\omega x0} \\ U_{\omega y0} \\ U_{\omega z0} \end{bmatrix} + \begin{bmatrix} n_{\omega x} \\ n_{\omega y} \\ n_{\omega z} \end{bmatrix} \quad (2.25)$$

Калібрування кутових швидкостей відбувається на поворотному столі. Датчик встановлюється так, щоб вісь чутливості біла паралельна осі обертання поворотного столу. Для більшої точності побудови статичної характеристики рекомендовано виконувати для декількох значень величин кутової швидкості, як мінімум для двох. Також виконуючи кожний вимір, сигнали гіроскопу осереднюються для суттєвого зменшення шумів вимірів, тому в подальших розрахунках вплив шумів враховувати не будемо. Для  $i$ -го виміру матимемо:

$$\begin{bmatrix} U_{\omega xi} \\ U_{\omega yi} \\ U_{\omega zi} \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \cdot \begin{bmatrix} \omega_{xi} \\ \omega_{yi} \\ \omega_{zi} \end{bmatrix} + \begin{bmatrix} U_{\omega x0}^* \\ U_{\omega y0}^* \\ U_{\omega z0}^* \end{bmatrix}. \quad (2.26)$$

Для подальших вимір зручно об'єднати 9 коефіцієнтів та 3 зміщення нуля в одну матрицю перетворивши матричне рівняння у вигляд :

$$\begin{bmatrix} U_{\omega xi} \\ U_{\omega yi} \\ U_{\omega zi} \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & U_{\omega x0}^* \\ r_{21} & r_{22} & r_{23} & U_{\omega y0}^* \\ r_{31} & r_{32} & r_{33} & U_{\omega z0}^* \end{bmatrix} \cdot \begin{bmatrix} \omega_{xi} \\ \omega_{yi} \\ \omega_{zi} \\ 1 \end{bmatrix}. \quad (2.27)$$

Спочатку задається обертання блоку гіроскопу навколо осі  $x$ . В цьому випадку  $\omega_{yi} = 0$ ,  $\omega_{zi} = 0$  та його можна спростити:

$$\begin{bmatrix} U_{\omega xi} \\ U_{\omega yi} \\ U_{\omega zi} \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & U_{\omega x0}^* \\ r_{21} & r_{22} & r_{23} & U_{\omega y0}^* \\ r_{31} & r_{32} & r_{33} & U_{\omega z0}^* \end{bmatrix} \cdot \begin{bmatrix} \omega_{xi} \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} r_{11} & U_{\omega x0}^* \\ r_{21} & U_{\omega y0}^* \\ r_{31} & U_{\omega z0}^* \end{bmatrix} \cdot \begin{bmatrix} \omega_{xi} \\ 1 \end{bmatrix}. \quad (2.28)$$

Оскільки при калібруванні задається  $N$  значень кутових швидкостей  $\omega_{xi}$ , отримуємо



$$\begin{bmatrix} U_{\omega x1} \dots U_{\omega xN} \\ U_{\omega y1} \dots U_{\omega yN} \\ U_{\omega z1} \dots U_{\omega zN} \end{bmatrix} = \begin{bmatrix} r_{11} & U_{\omega x0}^* \\ r_{21} & U_{\omega y0}^* \\ r_{31} & U_{\omega z0}^* \end{bmatrix} \cdot \begin{bmatrix} \omega_{x1} \dots \omega_{xN} \\ 1 \dots 1 \end{bmatrix}. \quad (2.29)$$

Такі ж матричні рівняння отримуємо і для обертів навколо осі у, і для обертів навколо осі z:

$$\begin{bmatrix} U_{\omega xi} \\ U_{\omega yi} \\ U_{\omega zi} \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & U_{\omega x0}^* \\ r_{21} & r_{22} & r_{23} & U_{\omega y0}^* \\ r_{31} & r_{32} & r_{33} & U_{\omega z0}^* \end{bmatrix} \cdot \begin{bmatrix} 0 \\ \omega_{yi} \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} r_{12} & U_{\omega x0}^* \\ r_{22} & U_{\omega y0}^* \\ r_{32} & U_{\omega z0}^* \end{bmatrix} \cdot \begin{bmatrix} \omega_{yi} \\ 1 \end{bmatrix}, \quad (2.30)$$

$$\begin{bmatrix} U_{\omega xN+1} \dots U_{\omega x2N} \\ U_{\omega yN+1} \dots U_{\omega y2N} \\ U_{\omega zN+1} \dots U_{\omega z2N} \end{bmatrix} = \begin{bmatrix} r_{12} & U_{\omega x0}^* \\ r_{22} & U_{\omega y0}^* \\ r_{32} & U_{\omega z0}^* \end{bmatrix} \cdot \begin{bmatrix} \omega_{y1} \dots \omega_{yN} \\ 1 \dots 1 \end{bmatrix}, \quad (2.31)$$

$$\begin{bmatrix} U_{\omega xi} \\ U_{\omega yi} \\ U_{\omega zi} \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & U_{\omega x0}^* \\ r_{21} & r_{22} & r_{23} & U_{\omega y0}^* \\ r_{31} & r_{32} & r_{33} & U_{\omega z0}^* \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ \omega_{zi} \\ 1 \end{bmatrix} = \begin{bmatrix} r_{13} & U_{\omega x0}^* \\ r_{23} & U_{\omega y0}^* \\ r_{33} & U_{\omega z0}^* \end{bmatrix} \cdot \begin{bmatrix} \omega_{zi} \\ 1 \end{bmatrix}, \quad (2.32)$$

$$\begin{bmatrix} U_{\omega x2N+1} \dots U_{\omega x3N} \\ U_{\omega y2N+1} \dots U_{\omega y3N} \\ U_{\omega z2N+1} \dots U_{\omega z3N} \end{bmatrix} = \begin{bmatrix} r_{13} & U_{\omega x0}^* \\ r_{23} & U_{\omega y0}^* \\ r_{33} & U_{\omega z0}^* \end{bmatrix} \cdot \begin{bmatrix} \omega_{z1} \dots \omega_{zN} \\ 1 \dots 1 \end{bmatrix}. \quad (2.33)$$

Рівняння (2.29), (2.31), (2.33) є лінійними рівняннями виду:

$$B = X \cdot A, \quad (2.34)$$

де А та В – відомі матриці заданих кутових швидкостей та сигналів гіроскопів відповідно, X – невідома матриця, яка складається з 6 невідомих параметрів моделі гіроскопів.

Рішення даного матричного рівняння по матриці напрямних косинусів має вигляд:

$$X = B \cdot A^+ = B \cdot A^T \cdot (A \cdot A^T)^{-1}. \quad (2.35)$$

Використавши рішення (2.35) до рівнянь (2.29), (2.31), (2.33) визначаються шукані параметри моделі вихідних сигналів гіроскопа:

$$\begin{bmatrix} r_{11} & U_{\omega x 0}^* \\ r_{21} & U_{\omega y 0}^* \\ r_{31} & U_{\omega z 0}^* \end{bmatrix} = \begin{bmatrix} U_{\omega x 1} \dots U_{\omega x N} \\ U_{\omega y 1} \dots U_{\omega y N} \\ U_{\omega z 1} \dots U_{\omega z N} \end{bmatrix} \cdot \begin{bmatrix} \omega_{x1} & 1 \\ \vdots & \vdots \\ \omega_{xN} & 1 \end{bmatrix} \cdot \left( \begin{bmatrix} \omega_{x1} & \dots & \omega_{xN} \\ 1 & \dots & 1 \end{bmatrix} \cdot \begin{bmatrix} \omega_{x1} & 1 \\ \vdots & \vdots \\ \omega_{xN} & 1 \end{bmatrix} \right)^{-1} \quad (2.36)$$

$$\begin{bmatrix} r_{12} & U_{\omega x 0}^* \\ r_{22} & U_{\omega y 0}^* \\ r_{32} & U_{\omega z 0}^* \end{bmatrix} = \begin{bmatrix} U_{\omega x N+1} \dots U_{\omega x 2N} \\ U_{\omega y N+1} \dots U_{\omega y 2N} \\ U_{\omega z N+1} \dots U_{\omega z 2N} \end{bmatrix} \cdot \begin{bmatrix} \omega_{y1} & 1 \\ \vdots & \vdots \\ \omega_{yN} & 1 \end{bmatrix} \cdot \left( \begin{bmatrix} \omega_{y1} & \dots & \omega_{yN} \\ 1 & \dots & 1 \end{bmatrix} \cdot \begin{bmatrix} \omega_{y1} & 1 \\ \vdots & \vdots \\ \omega_{yN} & 1 \end{bmatrix} \right)^{-1} \quad (2.37)$$

$$\begin{bmatrix} r_{13} & U_{\omega x 0}^* \\ r_{23} & U_{\omega y 0}^* \\ r_{33} & U_{\omega z 0}^* \end{bmatrix} = \begin{bmatrix} U_{\omega x 2N+1} \dots U_{\omega x 3N} \\ U_{\omega y 2N+1} \dots U_{\omega y 3N} \\ U_{\omega z 2N+1} \dots U_{\omega z 3N} \end{bmatrix} \cdot \begin{bmatrix} \omega_{z1} & 1 \\ \vdots & \vdots \\ \omega_{zN} & 1 \end{bmatrix} \cdot \left( \begin{bmatrix} \omega_{z1} & \dots & \omega_{zN} \\ 1 & \dots & 1 \end{bmatrix} \cdot \begin{bmatrix} \omega_{z1} & 1 \\ \vdots & \vdots \\ \omega_{zN} & 1 \end{bmatrix} \right)^{-1} \quad (2.38)$$

Розраховані коефіцієнти формуються у матриці:

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}, \quad (2.39)$$

$$U_{\omega 0} = \begin{bmatrix} U_{\omega x 0} \\ U_{\omega y 0} \\ U_{\omega z 0} \end{bmatrix}. \quad (2.40)$$

Вирахування дійсних значень сигналу блоку акселерометрів виконується за формулою:

$$A = K^{-1}(U_a - U_{a0}), \quad (2.41)$$

а з блоку гіроскопів за формулою:

$$W = R^{-1}(U_{\omega} - U_{\omega 0}). \quad (2.42)$$

Для зменшення обчислювального навантаження у програмний код вписуються завчасно інвертовані значення масштабних. Таким чином для використаного датчика матриці калібрувальних коефіцієнтів мають наступний вигляд:

для акселерометрів:

$$K^{-1} = \begin{bmatrix} -0,0153 & -1,0029 & -0,007 \\ 0,9927 & -0,0279 & 0,0519 \\ -0,0603 & -0,0038 & 0,9879 \end{bmatrix}, \quad (2.43)$$

$$U_{a0} = \begin{bmatrix} -0,0495 \\ 0,1026 \\ 0,4091 \end{bmatrix}; \quad (2.44)$$

для гіроскопів:

$$R^{-1} = \begin{bmatrix} 0,0171 & -0,0002 & -0,0009 \\ -0,0002 & 0,0175 & 0,0002 \\ 0,0012 & 0,0378 & 0,0175 \end{bmatrix}, \quad (2.45)$$

$$U_{\omega 0} = \begin{bmatrix} 0,0342 \\ 0,0377 \\ 0,0008 \end{bmatrix}. \quad (2.46)$$

### *Програмне забезпечення*

Як було зазначено у вступі такі два пакети для Matlab як Arduino Support Package for Matlab та Simulink Support Package for Arduino Hardware дозволяють програмувати плати Arduino зокрема й Arduino Uno у програмному самому програмному середовищі. В обох пакетах є базові функції для роботи з платою, проте є декілька відмінностей. Так пакет Arduino Support Package for Matlab не дає можливості вивантажувати написану програму на саму плату, а лише працювати із нею в режимі реального часу. А пакет Simulink Support Package for Arduino Hardware дозволяє як працювати з платою в режимі реального часу, так і виконувати вивантаження програми. Також варто відмітити особливість створення алгоритму у Simulink – за певний функцію відповідає окремий блок, проте залишається можливим вставлення скриптової частини у вигляді окремого блоку. В залежності від пакету розрізняється створення зв'язку та налаштування плати для роботи.

Для випадку Matlab налаштування плати виконується в самому скрипті шляхом створення об'єкту плати. Базові функції як: конфігурування контактів на

вхід або вихід, зчитування аналогового сигналу, тощо, виконується функціями у відношенні до об'єкту плати. Також у скрипті виконується налаштування інших пристроїв для взаємодії, наприклад створення об'єктів сервомашин, що використовуються в пакеті датчиків тощо. Якщо потребується виконувати певні інструкції необмежену кількість часу можна прописувати їх у нескінченний цикл, який буде виконуватись до кінця розриву з'єднання, або примусовим припиненням з боку користувача.

При використанні Simulink налаштування плати виконується в налаштуваннях моделі. Для базових функцій в бібліотеці блоків є відповідні блоки. Для специфічних та виокремлених пристроїв (сервомашини, датчики) існують окремі блоки. Виконання створеного алгоритму відбувається від початку до кінця моделі циклічно протягом часу моделювання, або нескінченно за умови вивантаження алгоритму на плату.

Обидва пакети самі виконують перетворення байт-коду датчиків у значення прискорення в розмірності  $\text{м/с}^2$  та кутової швидкості – град/с. Проте таке калібрування має свої неточності тому виникає необхідність у проведенні додаткового калібрування значень прискорення та кутової швидкості. Калібрування виконується за формулами (2.41) та (2.42) з використанням матриць калібрування (2.43), (2.44), (2.45), (2.46).

Розглянемо алгоритм інклінометру виконаний в обох пакетах.

### **Виконання алгоритму інклінометру за допомогою пакету Arduino Support Package for Matlab**

Написання та виконання програми починається з визначення констант – коефіцієнтів переведення з радіан у градуси та навпаки, кроку опитування датчика, матриць калібрування.

Далі створюється об'єкт плати командою  $a = \text{arduino}('COM3', 'UNO')$ , де «a» - назва створюемого об'єкту плати, «COM3» - порт до якого під'єднана плата, «UNO» - назва плати для якої встановлюється зв'язок. Надалі необхідно створити об'єкти сервомашин та інерціального вимірювального модуля.

Для сервомашин команда створення об'єкту виглядає наступним чином: `s1 = servo(a,'D9');` `s2 = servo(a,'D10');` де «s1» та «s2» - назва створюємих об'єктів сервомашин, «a» - об'єкт плати, для якої створюється з'єднання, «D9» та «D10» цифрові контакти з властивістю широтно-імпульсного модулятора.

Створення об'єкту для IBM виглядає наступним чином: `imu = mpu6050(a);` де «imu» - назва створюємого об'єкту IBM, «a» - об'єкт плати, для якої створюється з'єднання. За умови наявності тулбоксу «Sensor Fusion and Tracking Toolbox™ and Navigation Toolbox™» створення налаштування об'єкту IBM розширюється і додається можливість задання частоти опитування датчика, кількість знятих відліків за зчитування, режим зчитування, вихідний формат даних, часовий формат записування даних. Наприклад, команда для зчитування одного значення з датчика з частотою 100 відліків за секунду матиме вигляд: `imu=mpu6050(a,'SampleRate',100,'SamplesPerRead',1);`

Після закінчення налаштувань основних об'єктів взаємодії переходимо до створення нескінченного циклу. Для початку необхідно зчитати значення кутових швидкостей та прикорень з датчика. Це виконується за допомогою команд наступних команд, вихідні значення записуються у вигляді матриці розміром 3x1.

Надалі зчитані для зчитаних даних проводять калібрування, сигнали гіроскопів записуються до кососиметричної матриці. Для сигналу акселерометра виконується обчислювання кутів відносно вектору вільного падіння, з яких формується МНК для акселерометрів. Для гіроскопів виконується чисельне інтегрування методом прямокутників та обрахування нового положення рівнянням Пуассона. Отримані МНК пропускаються через рівняння комплементарного фільтру результатом якого є МНК з якої вираховуються кути  $\varphi$ ,  $\theta$ ,  $\psi$ .

Двоє з цих кутів є шуканими і є кутам крену та тангажу, що показують нахил приладу, а як наслідок і об'єкту, на якому встановлено прилад. Відобразити ці кути можна за допомогою положення показників, які закріплені на валу сервомашин, Дана процедура виконується за допомогою команди `writePostition(s1,ang2serv(Psi)); writePostition(s2,ang2serv(Tet));`, де функція «ang2serv()» - перерахування кута з градусів в позицію сервомашини від 0 до 1 (0

– крайнє лівє положення, 1 – крайнє правє положення), «Psi» та «Tet» - значення кутів крену та тангажу.

Повний текст коду наведено у додатку А.

### Виконання алгоритму інклінометру за допомогою пакету Simulink Support Package for Arduino Hardware

Створення моделі алгоритму для інклінометру у середовищі Simulink має дещо інший вигляд. Оскільки основними компонентами даного середовища є блоки, то всі операції виконуються саме в них. Повна схема алгоритму наведена на рис. 2.3.

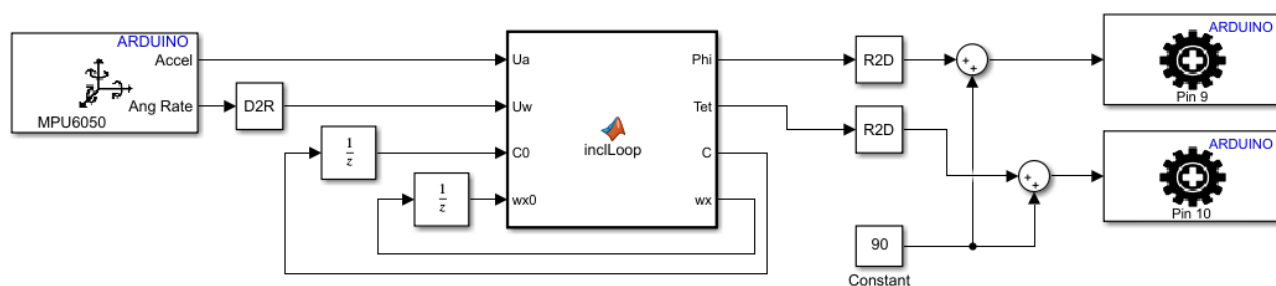


Рисунок 2.3 Схема алгоритму інклінометру у Simulink

Етапи налаштування моделі та створення даної схеми для плати Arduino Uno виглядають наступним чином. Перш за все необхідно в налаштуваннях моделі вибрати плату, з якою буде вестись робота. Дана процедура виконується через вкладку «HARDWARE» на панелі інструментів. В розділі «HARDWARE BOARD» обирається пункт «Select Another Hardware Board». Після відкриття вікна конфігурації параметрів, у вкладці «Hardware Implementation», зі списку «Hardware board» обирається бажану плату - «Arduino Uno». Інші налаштування можна залишити стандартними (рис. 2.4).

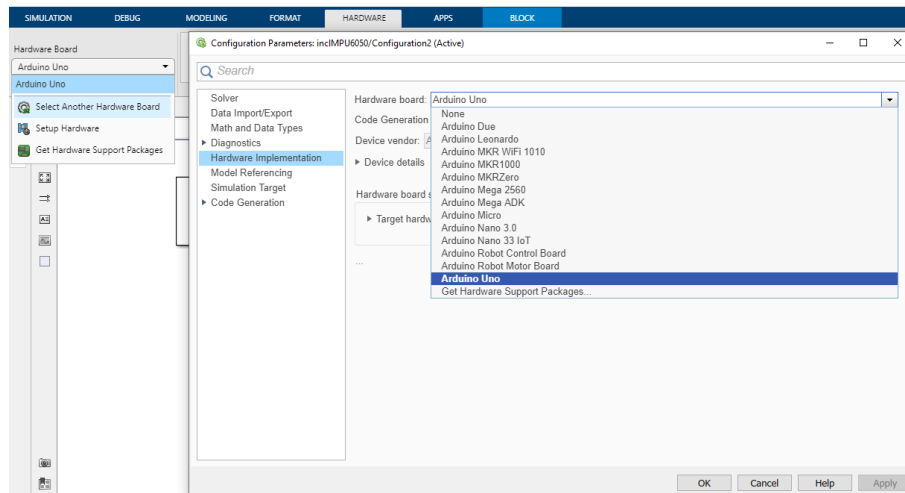


Рисунок 2.4. Вибір бажаної плати для створення моделі

Надалі, з браузера бібліотеки, у вкладці «Simulink Support Package for Arduino Hardware -> Sensors», перетягується в робочий простір необхідні блок «MPU6050 IMU Sensor». Двічі клацнувши по ньому відкривається вікно налаштування вихідних параметрів датчика. У даному вікні доступні для змін такі параметри:

- адреса пристрою:
  - 0x68;
  - 0x69;
- вибір даних, що будуть виводитись блоком:
  - прискорення (м/с<sup>2</sup>);
  - кутова швидкість (град/с);
  - температура (°C);
  - статус.
- діапазон вимірювань акселерометрів:
  - 2 g;
  - 4 g;
  - 8 g;
  - 16 g;
- діапазон вимірювань гіроскопів:

- 250 град/с;
- 500 град/с;
- 1000 град/с;
- 2000 град/с;
- полоса пропускання акселерометрів та гіроскопів;
- крок видачі інформації (на вибір користувача, за промовчуванням складає -1).

Для роботи інклінометру необхідні значення акселерометрів та гіроскопів, тому обираються для виводу лише вони. Діапазон вимірювань для акселерометрів задається 4 g, для гіроскопів – 500 град/с, полосу пропускання акселерометрів та гіроскопів можна залишити стандартними. Крок видачі інформації встановлюється 0,01 с (100 Гц). Вихідні вектори даних мають розмірність 3x1 (рис 2.5).

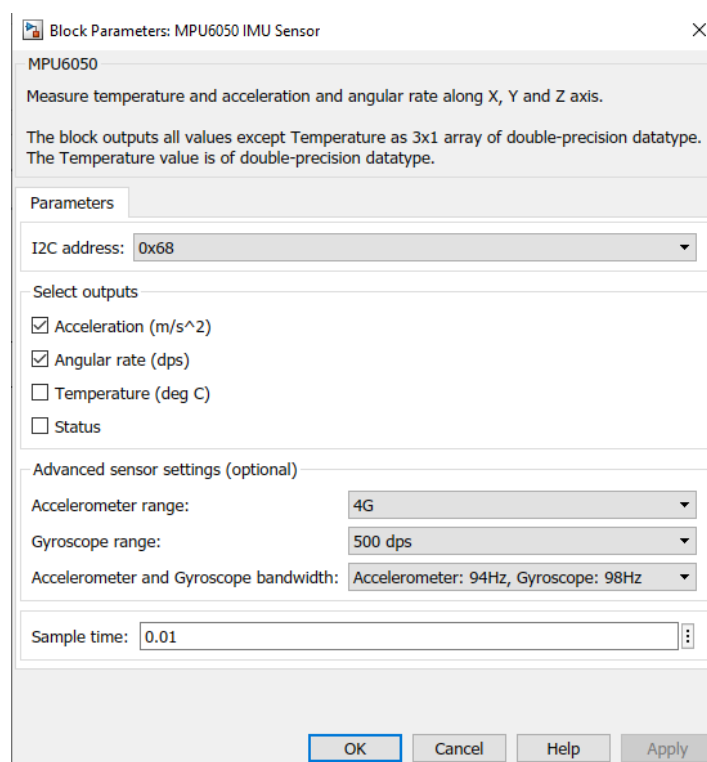


Рисунок 2.5 Вікно налаштування блоку «MPU6050 IMU Sensor»

Оскільки типовою розмірністю кутової швидкості в алгоритмі інклінометру є рад/с, а вихідне значення блоку датчика є град/с необхідно додати в ланцюг



блок, що перераховує значення град/с в рад/с - «Degrees to Radians». Знайти цей блок можливо у вкладці браузера бібліотеки блоків «Simulink Extras -> Transformations».

Надалі, над отриманими перетвореннями необхідно провести математичні перетворення, аналогічні до тих, що виконувались в підрозділі виконання алгоритму інклінометра в пакеті Arduino Support Package for Matlab. Для зручності ці перетворення можна залишити у вигляді тестового коду, а сам код, провівши певні зміни, вставити до спеціального блоку «MATLAB function» із вкладки «Simulink -> User-Defined Functions». Додавши цей блок до робочого поля, необхідно відкрити його і записати код алгоритму. Записаний код наведено у додатку А. Команда «`%#codegen`» необхідна для генерації С коду з MATLAB коду при вивантаженні програми на плату.

Зберігши зміни та повернувшись до робочого поля моделі можна і побачимо наступні зміни в зовнішньому вигляді блоку – в його лівій частині з'явилися чотири входи підписані як «`accel`», «`gyro`», «`C0`» та «`wx0`». В правій частині будуть виходи «`Tet`» - значення кута крену, «`Phi`» - значення кута тангажу, «`C`» - МНК, «`wx`» - кососиметрична матриця. На входи «`accel`» й «`gyro`» подаються значення акселерометрів та гіроскопів відповідно.

Виходи «`C`» та «`wx`» йдуть зворотнім зв'язком на входи «`C0`» та «`wx0`» відповідно, але через необхідність задання початкового значення стану системи, у зворотній зв'язок додається блок «Unit Delay», який можна знайти на вкладці «Simulink -> Discrete». Всередині цього блоку прописуються початкові умови у вигляді масиву. Запис початкових умов для МНК та кососиметричної матриці в «Unit Delay» наведено у Додатку А.

Значення із виходів «`Phi`» та «`Tet`» передаються далі. Для зручності зчитування вихідні значення крену і тангажу переводяться з радіан у градуси блоком «Radians to Degrees». Блок можна знайти у вкладці браузера бібліотеки блоків «Simulink Extras -> Transformations».

Для сервомашин існують стандартні блоки. Ці блоки знаходяться у вкладці «Simulink Support Package for Arduino Hardware -> Common». Необхідним блоком

є «Standard Servo Write». Для роботи інклінометра необхідно додати на робоче поле моделі два таких блоки. Відкривши налаштування блока, двічі натиснувши на нього, для одного задається параметр «Pin number» 9, для іншого – 10, що відповідають контактам з функцією ШІМ. Вхідне значення для цих блоків складає від 0 до 180, де 0 – положення валу сервомашини 0 градусів, 180 – положення валу сервомашини 180 градусів.

Зважаючи на те, що вихідне значення кутів матиме значення від -90 до 90 градусів як для крену, так і для тангажу, то вводиться коригуючий коефіцієнт в 90 одиниць який задається блоком «Constant» із вкладки «Simulink -> Sources». До значень крену і тангажу коригуючий коефіцієнт додається за допомогою блоку «Sum», який береться із вкладки «Simulink -> Math Operations». Скореговані значення направляються на блоки сервомашин.

Програма виконується циклічно за заданими напрямками стрілок.

### *Конструкція інклінометра*

#### **Компоненти інклінометра**

Для інклінометру можна виділити такі основні блоки:

- блок чутливих елементів;
- блок оброблювача інформації;
- блок живлення;
- блок відображення.

Для блоку чутливих елементів обрано інерціальний вимірювальний модуль (ІВМ, англ. – Inertial Measurement Unit) на базі МЕМС датчика MPU-6050 (рис. 2.6). Даний модуль складається з тривісного блоку акселерометрів та тривісного блоку гіроскопів. Для оцифрування сигналу акселерометрів та гіроскопів використовується 16-бітний аналого-цифровий перетворювач. Для датчиків можливий вибір діапазону вимірювань, для акселерометрів -  $\pm 2$ ,  $\pm 4$ ,  $\pm 8$  та  $\pm 16$  g, для гіроскопів  $\pm 250$ ,  $\pm 500$ ,  $\pm 1000$  та  $\pm 2000$  °/с. Комунікація з регістрами датчика відбувається на швидкості 400 кГц за допомогою інтерфейсу I2C. Сам МЕМС

датчик розташовується входить до складу плати GY-521. Напруга живлення складає 3,3 В, проте можливе підключення і до джерела у 5 В [14].

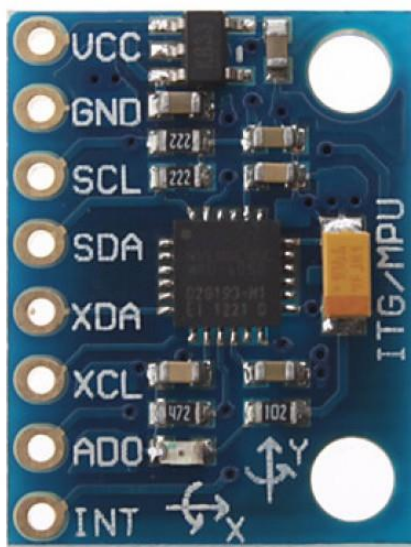


Рисунок 2.6 Плата GY-621 з чіпом MPU6050

Для блоку оброблювача інформації обрано плату Arduino Uno (рис. 2.7). Плата базується на мікроконтролері ATmega328, має 32 кБ флеш-пам'яті, 2 кБ оперативної пам'яті, тактова частота 16 МГц, 14 цифрових пінів, 6 з яких мають широтно-імпульсний модулятор, 5 аналогових входних пінів, 2 окремих піна для I2C, виходи для напруги в 5 та 3,3 В. Аналогом може слугувати інша плата Arduino – Arduino Nano, яка має приблизно такі ж основні характеристики та менший розмір [24].



Рисунок 2.7 Плата Arduino Uno

У блок живлення входить два акумулятору типу 18650 (рис. 2.8) та плата контролю заряду акумуляторів. .



Рисунок 2.8 Акумулятор типу 18650

Для блоку відображення обрано дві сервомашини НК 15178 (рис 2.9). Робоча напруга 4,8 ~ 6 В. Крутний момент: 1,2 кг/см при живленні від 4,8 В; 1,4 кг/см при живленні від 6 В. Швидкість: 0,1/60° при живленні від 4,8 В; 0,09/60° при живленні від 6 В. Робоча область 180 градусів. Штепсельна виделка: JR [25].



Рисунок 2.9 Сервомашини НК 15178

### Конструктивна схема інклінометра

Для взаємодії компонентів їх необхідно об'єднати в одну схему. Схема такого об'єднання матиме вигляд зображеного на рис. 2.10.

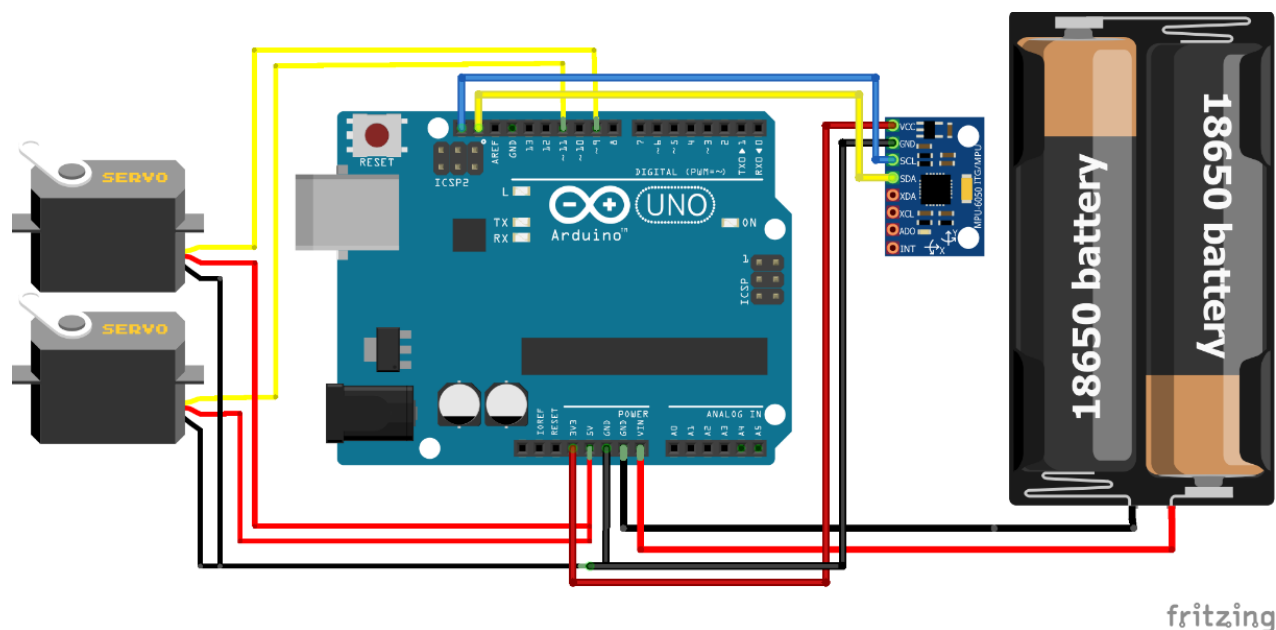


Рисунок 2.10 Конструктивна схема інклінометру

Живлення Arduino Uno здійснюється від двох акумуляторів 18650, що під'єднані контактом «+» до «Vin», а «-» – «GND». Самі акумулятори з'єднані з контролером заряду. На схемі дрід «+» позначений червоним кольором, а дрід «-» – чорним.

Оскільки чутливий елемент MPU6050 взаємодіє за інтерфейсом I2C, а на платі Arduino Uno існують спеціальні контакти для такого інтерфейсу «SDA» та «SCL», тому інформаційні контакти під'єднано датчика до них. «VCC», що відповідає за «+», під'єднаємо до «3.3V», а «GND», що відповідає за «-», до «GND» на платі. На схемі дрід «SDA» позначено жовтим кольором, дрід «SCL» – синім, дрід «VCC» – червоним, а дрід «GND» – чорним.

Для роботи сервомашини необхідно 3 контакти – «+», «-» та інформаційний. «+» приєднується до «5V», «-» до «GND». Зважаючи, що є аналоговим пристроєм, то необхідно, щоб на інформаційний вхід йому поступав аналоговий сигнал. Для цього інформаційний дрід необхідно під'єднати до контакту з широтно-імпульсним модулятором, який моделює аналоговий сигнал. Для вибраної плати такими контактами є: «D3», «D5», «D6», «D9», «D10» і «D11». В даному рішенні обрано контакти «D9» та «D10». На схемі інформаційний дрід позначено жовтим кольором, дрід «+» – червоним, а дрід «-» – чорним.

### Корпус інклінометра

Забезпечити мобільність приладу, а також його встановлення на будь-якій поверхні може корпус. Корпус повинен компонувати в собі елементи приладу, захищати їх від несприятливих зовнішніх подразнень, забезпечувати чіткі та ясні візуальні зчитування показань приладу.

Розроблений корпус, який зображений на рис. 2.11, складається з трьох частин: основи, передньої панелі та кришки.

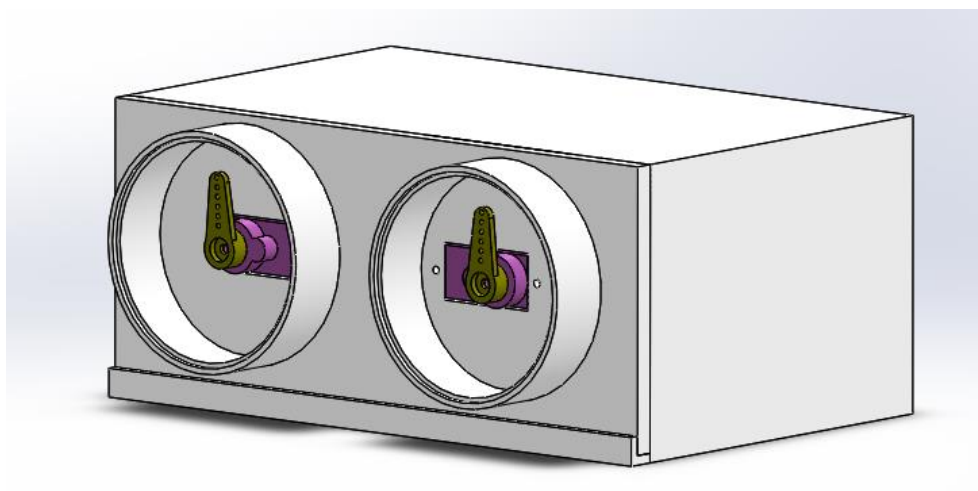


Рисунок 2.11 Зовнішній вигляд моделі інклінометру

На основі корпусу розташовуються та закріплюються основні компоненти приладу. На рисунку 2.12 зображено тривимірну модель основи корпусу. Посередині знаходиться Г-образна стійка для закріплення на ньому блоку чутливих елементів якомога ближче до геометричного центру інклінометра. Зверху Г-образної стойки існує жолоб для можливих напівсфер припою та вирівнювання закріплення плати. Ліворуч, від центру, знаходиться місце для плати Arduino. Праворуч - місце для акумуляторів. Знизу знаходиться жолоб для вставлення у нього передньої частини корпусу. По боковому та задньому краю основи залишено місце для приєднання кришки. До основи компоненти кріпляться гвинтами.

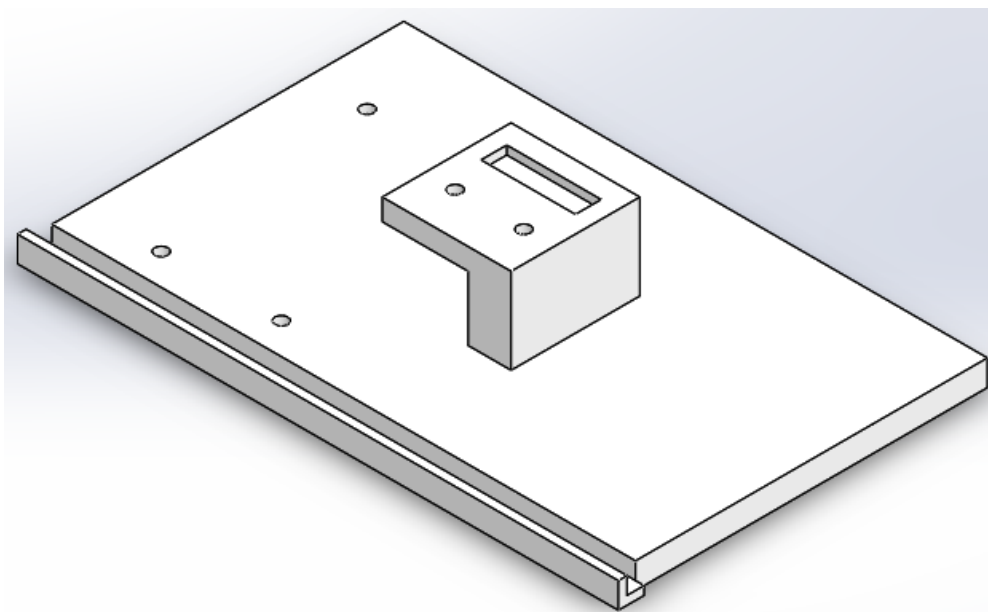


Рисунок 2.12 Модель основи корпусу інклінометра

Передня панель, тривимірна модель якої зображена на рис. 2.13, має два місця для закріплення сервомашин. Задля захисту показчиків сервомашин, від зовнішніх механічних впливів, створено два кільцевих виступи всередині яких знаходяться сервомашини. На вершині виступів утворено жолоб для захисного скла з циферблатом. Кріпляться до основної частини гвинтами.

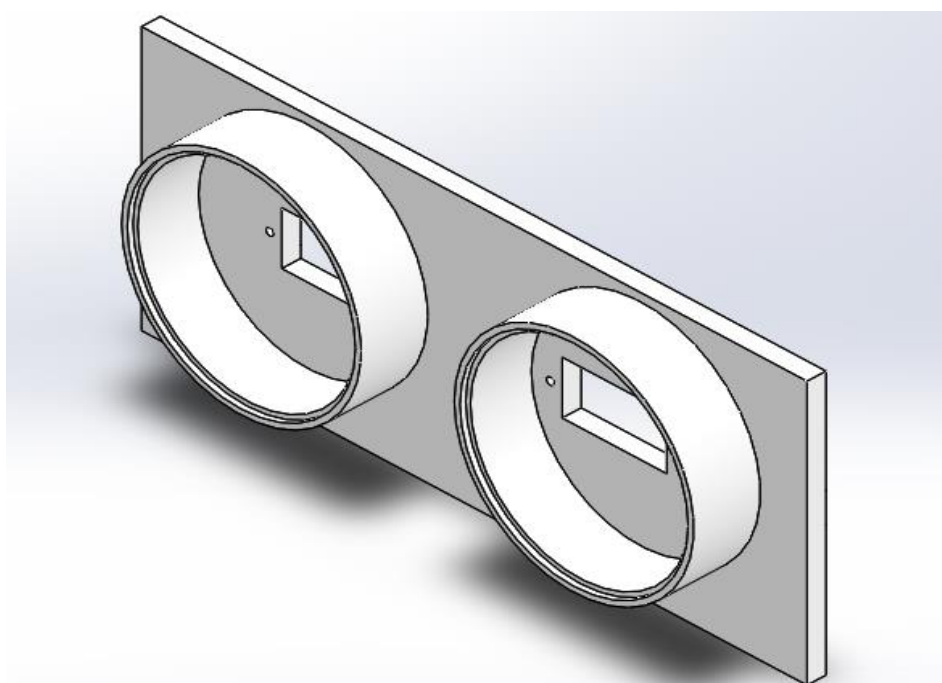


Рисунок 2.13 Модель передньої панелі корпусу інклінометра



Кришка корпусу (рис. 2.14) покриває задній, верхній та бокові поверхні приладу, прилягаючи до основи та передньої панелі й закріплюється до них гвинтами.

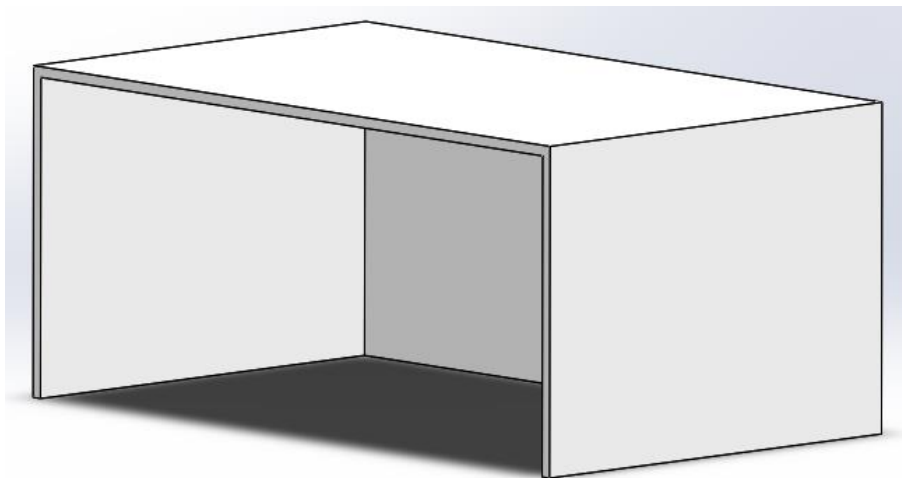


Рисунок 2.14 Модель кришки корпусу інклінометра

Зважаючи на те, що виробництво інклінометра не носить масовий або серійний характер, виготовлення корпусу краще виготовити методом 3D друку. Забезпечення встановлення приладу на поверхні виконується за допомогою двосторонньої клейкої стрічки.

### *Висновки до розділу 2*

В цьому розділі було розглянуто існуючі форми кінематичних рівняння орієнтації кутового положення тіла, можливі варіанти створення алгоритму БІСО, методи чисельного інтегрування, початкова виставка БІСО та калібрування чутливих елементів. Для задачі створення інклінометру вибралась такі елементи алгоритму: в якості кінематичного рівняння – рівняння Пуассона, яке використовує найбільш зручний та наочний кінематичний параметр; для чисельного інтегрування рівняння Пуассона вибрався найпростіший метод першого порядку – лівих прямокутників; для компенсації дрейфів мікромеханічних гіроскопів обрано найпростіший серед усіх фільтрів – комплементарний фільтр; приведено алгоритм початкової виставки системи та калібрування.



Також розглянуто створення алгоритму інклінометру у програмному середовищі Matlab за допомогою пакетів Arduino Support Package for Matlab та Simulink Support Package for Arduino Hardware. Алгоритм створений з використанням пакету Arduino Support Package for Matlab дозволяє роботу приладу лише при підключенні до самого Matlab без вивантаження програми на плату Arduino. Пакет Simulink Support Package for Arduino Hardware дозволяє вивантажувати розроблений алгоритм на плату Arduino для подальшої автономної роботи системи. Налаштування роботи з ЧЕ та сервомашинами виконується за допомогою спеціальних функцій для Matlab, або за допомогою блоків для Simulink. Для Matlab налаштування ЧЕ виконується у вигляді встановлення частоти опитування датчика та кількості зчитаних відліків. У Simulink виконується не тільки налаштування частоти опитування системи, а й полоса пропускання і діапазон вимірювання датчиків. Величина вихідних даних для акселерометрів –  $\text{м/с}^2$ , для гіроскопів –  $\text{град/с}$ . Калібрування виконується пакетами автоматично, проте, через не достатню точність даних проводиться до калібрування. Величина кутової швидкості для алгоритму складає –  $\text{рад/с}$ , тому кутова швидкість перераховується з  $\text{град/с}$  у  $\text{рад/с}$  відповідним коефіцієнтом або блоком. Після вирахування необхідних МНК значення кутів крену та тангажу перераховується у градуси. Значення кута передаються на сервомашини для їх відображення положенням стрілки.

Для інклінометра було розглянуто його складові компоненти та їх короткі технічні характеристики. В якості ЧЕ обрано МЕМС датчик MPU6050, в склад датчика входить блок ММГ та ММА. В якості обчислювального приладу вибрано плату Arduino Uno. В якості показчиків вирахованого приладом кута вибрано дві сервомашини НК15178. Живлення приладу здійснюється за допомогою двох акумуляторів типу 18650, перезарядження котрих виконується за допомогою плати контролю заряду. Для мобільності та захисту компонентів приладу розроблено корпус. Виготовлення корпусу здійснюється методом 3D друку. Для кріплення до поверхонь використовується двостороння клейка стрічка.



### Розділ 3 Моделювання інклінометру

Величина коефіцієнту комплементарного фільтру для формули (2.8) впливає на достовірність та швидкість відображення інформації о кутах крену та тангажу тому важливо вибрати оптимальний коефіцієнт для датчиків приладу. Так за великого значення коефіцієнту на покази приладу почнуть впливати сторонні прискорення, що фіксують акселерометри, занадто малий – знижується точність та швидкодія вираховування й відображення кутів. Визначення коефіцієнту фільтру виконується емпіричним методом на основі тестових записів показань датчиків.

Вибір величини коефіцієнта  $k$  виконувався за наступними експериментальними даними. Для перевірки дії коефіцієнту для рухомої системи над блоком чутливих елементів виконувались такі серія дій навколо осі  $y$ : стан спокою, повільний нахили до 90 градусів в обидва боки навколо осі, стан спокою, нахили в обидва боки з середньою швидкістю, стан спокою, швидкі нахили в обидві, стан спокою, імітування дії лінійних прискорень на рухи датчику у площині, стан спокою, випадкові нахили датчика, стан спокою. Графіки сигналів датчиків зображено на рис 3.1 та рис. 3.2.

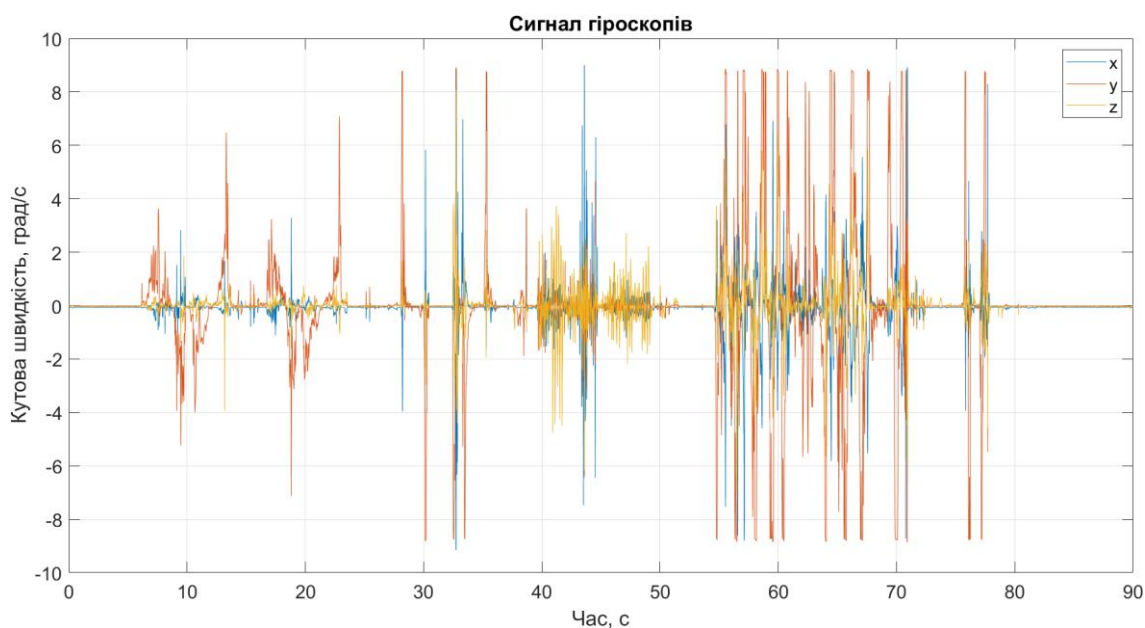


Рисунок 3.1 Сигнали кутової швидкості для тестових переміщень

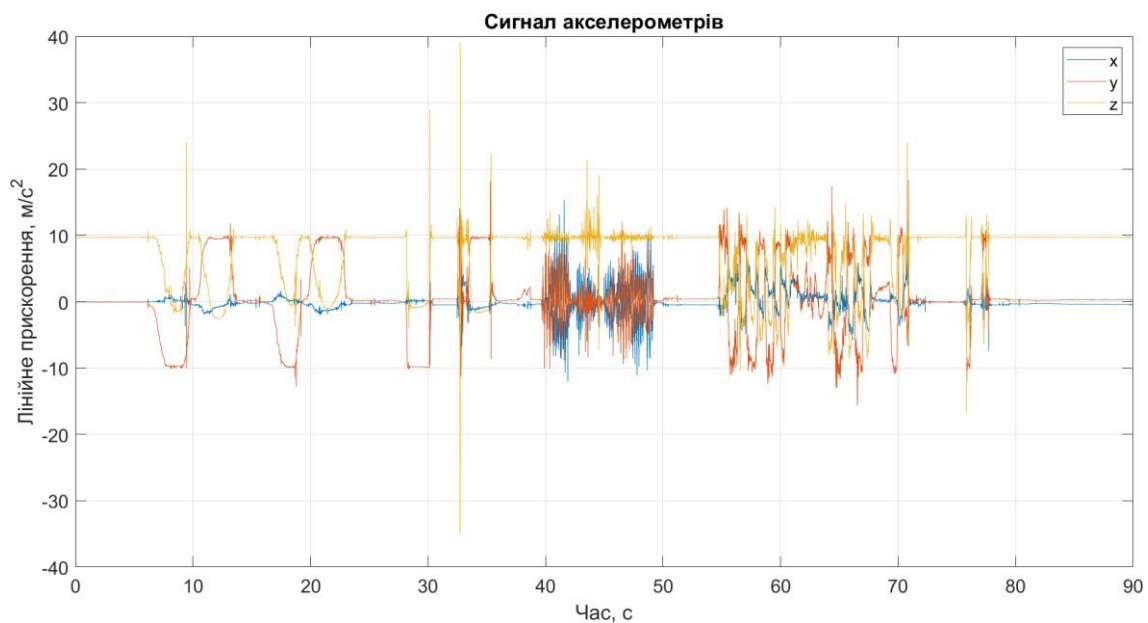


Рисунок 3.2 Сигнали лінійного прискорення для тестових переміщень

Для перевірки системи в статичному режимі записано сигнал датчиків, які були залишені на горизонтальній площині протягом тривалого періоду часу. На рис. 3.3 та рис 3.4 показано сигнали чутливих елементів записаний протягом 72 хв.

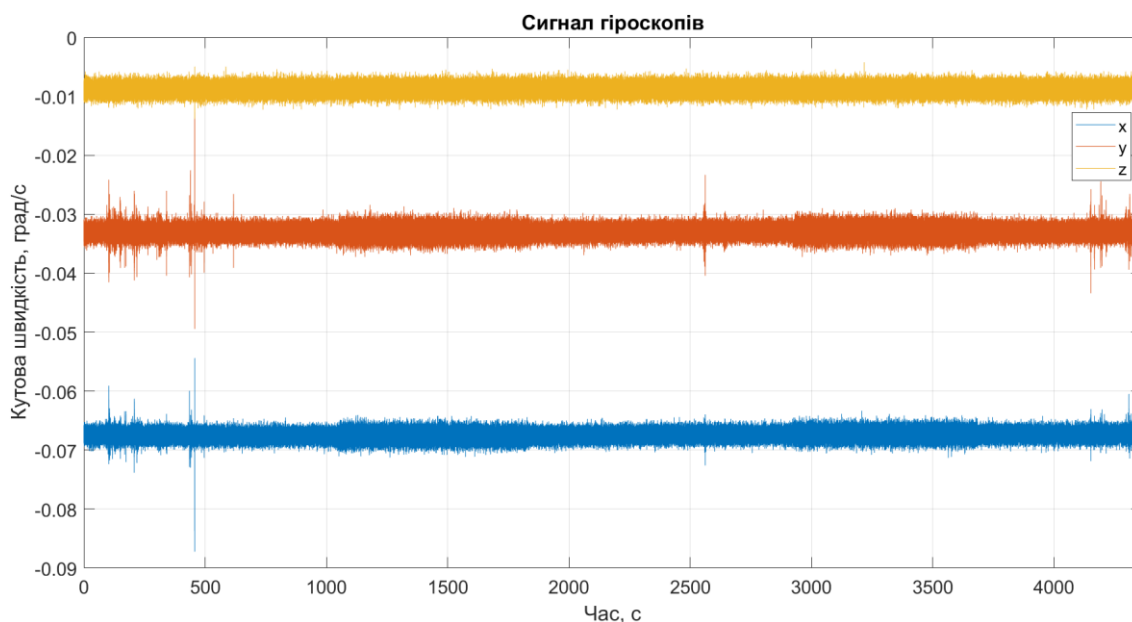


Рисунок 3.3 Сигнали кутової швидкості для статичних випробувань

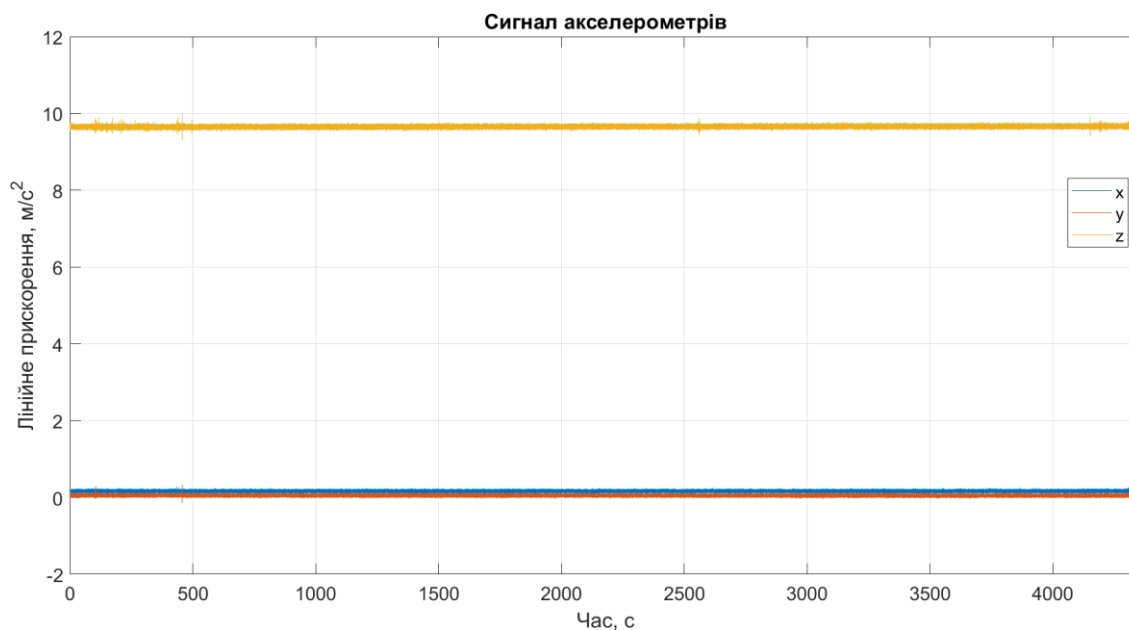


Рисунок 3.4 Сигнали лінійного прискорення для статичного випробування

Для першого тесту виберемо такі коефіцієнти 0,9, 0,5, 0,1, 0,01. Результати роботи фільтру з такими коефіцієнтами зображені на рис. 3.5, рис. 3.6, рис. 3.7 та рис.3.8. Для коефіцієнтів 0,9 та 0,5 спостерігається висока зашумленість вирахуваних кутів. На етапі дії лінійного прискорення (40 – 50 секунда) в одній площині спостерігається великі й часті зміни вирахованого кутового положення, що не є дійсних для даних зображених на рис. 3.1 і рис.3.2. В той же час значення коефіцієнту в 0,01 показує, що нахили не відбувались на 90 градусів, а значно менше, плавність графіків свідчить про низьку швидкодію системи, що не відповідає дійсності та не є прийнятним. Серед розглянутих значень, коефіцієнт 0,1 демонструє швидкість зміни кута, що схоже на реальні нахили, вихідне значення кутів позбавлене стрибків, які викликані діями прискорень на датчики та при дії лінійного прискорення не великих змін значення кута. Таким чином, серед розглянутих значень – коефіцієнт 0,1 є найбільш сприйнятним для використання.

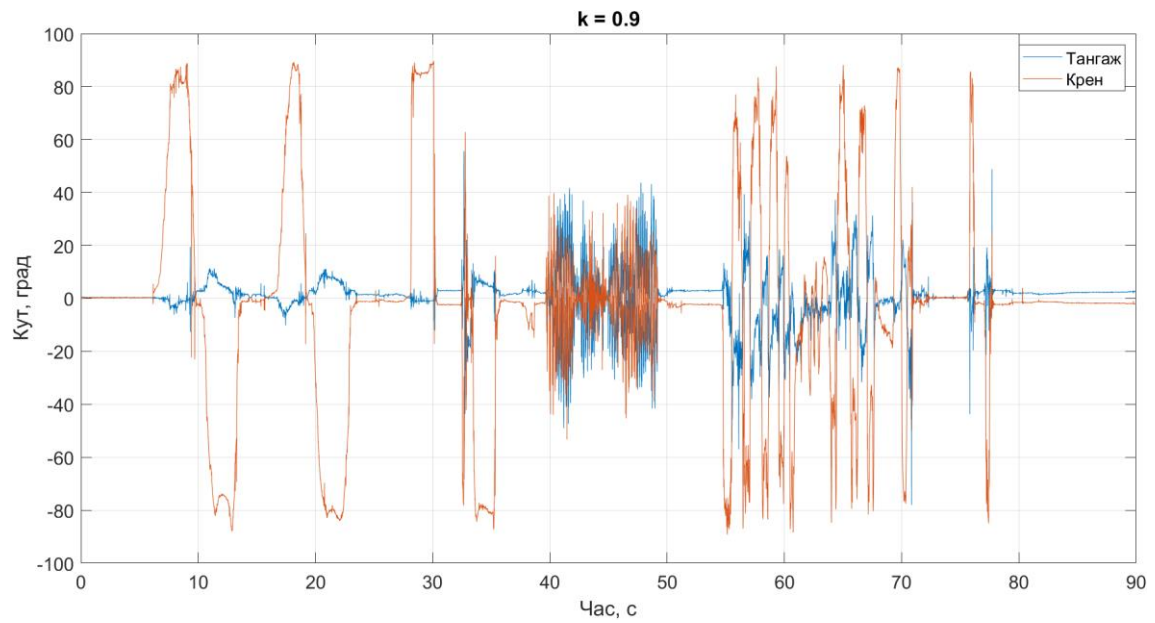


Рисунок 3.5 Результат роботи фільтру для коефіцієнта 0,9

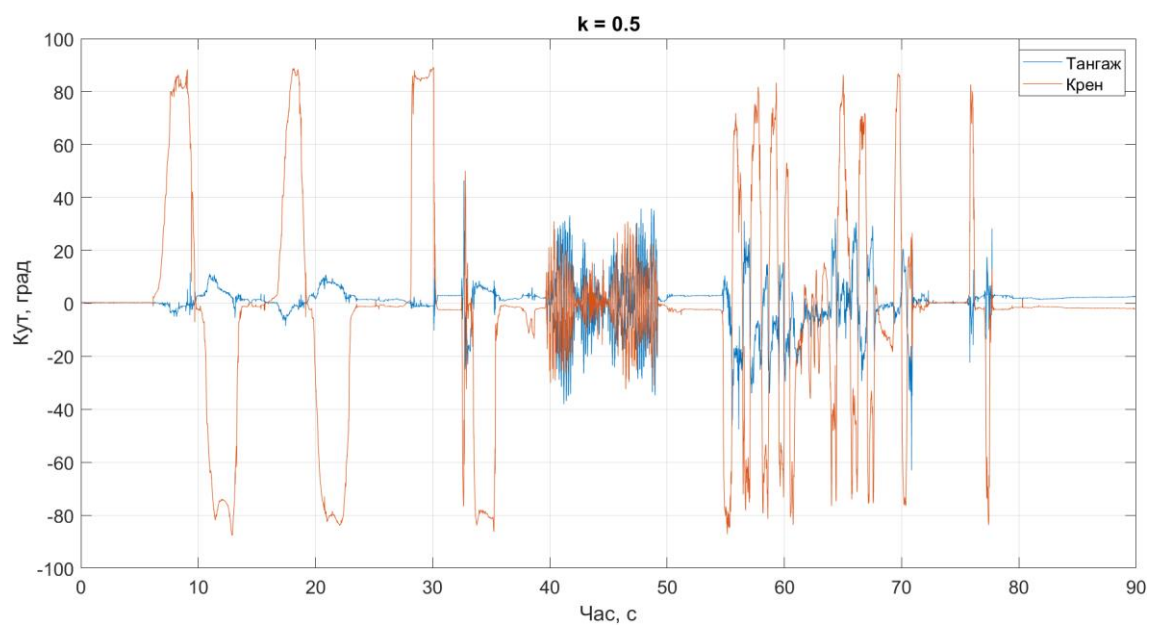


Рисунок 3.6 Результат роботи фільтру для коефіцієнта 0,5

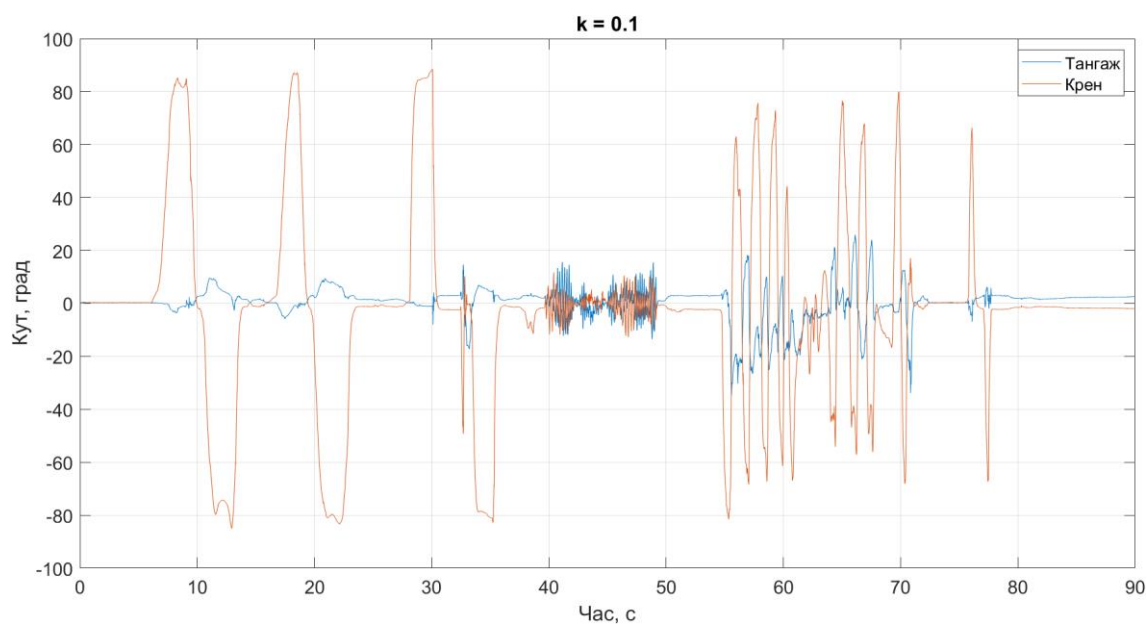


Рисунок 3.7 Результат роботи фільтру для коефіцієнта 0,1

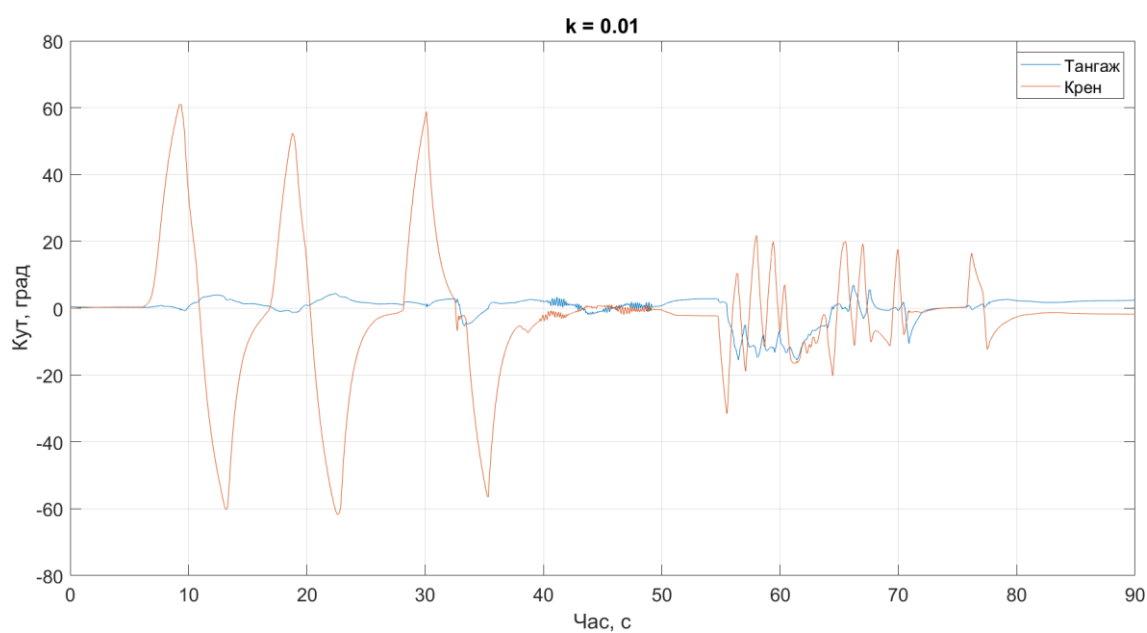


Рисунок 3.8 Результат роботи фільтру для коефіцієнта 0,01

Для підвищення якості та для впевненості у перевазі коефіцієнта 0,1 розглянемо найближчі сусідні до нього коефіцієнти: 0,15 та 0,05. Результат використання цих коефіцієнтів зображено на рис. 3.9 та рис. 3.10 відповідно. В порівнянні з 0,1 коефіцієнт 0,15 має більшу чутливість до прискорень з акселерометрів, що виражається у більшій нестабільності показань кутів та

чутливості до прикорень. В свою чергу коефіцієнт 0,05 має низьку швидкість відображення кутів.

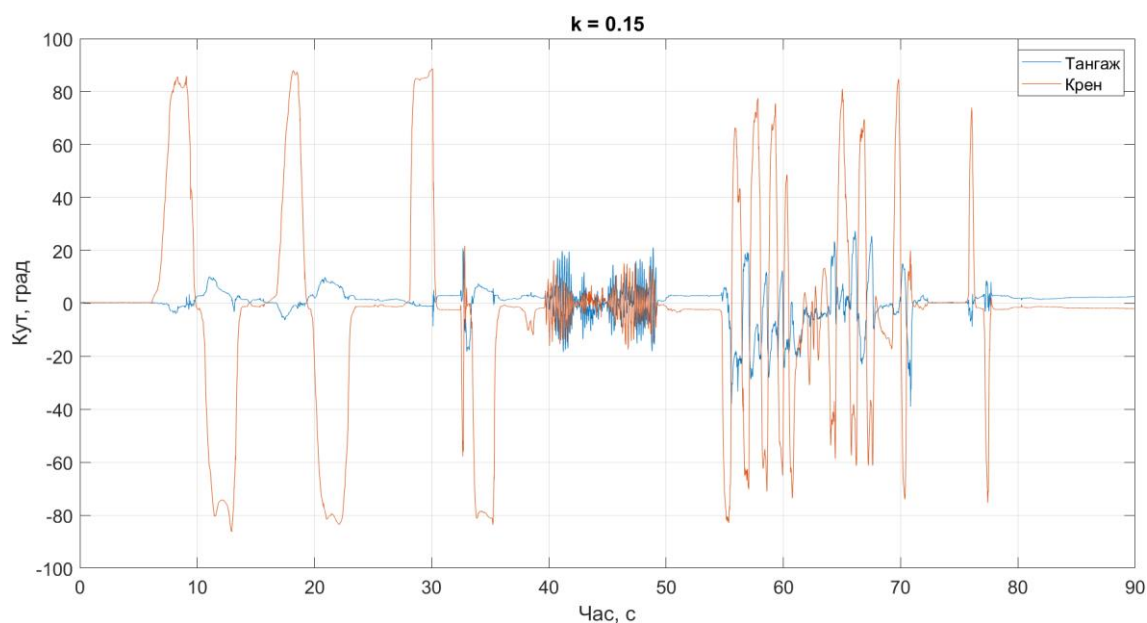


Рисунок 3.9 Результат роботи фільтру для коефіцієнта 0,15

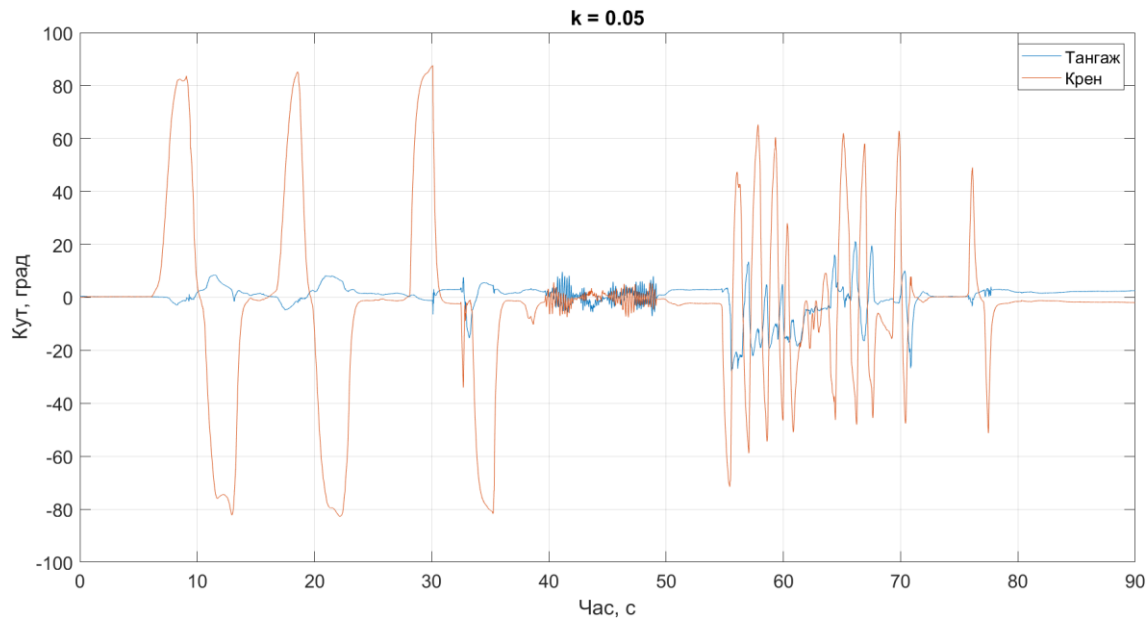


Рисунок 3.10 Результат роботи фільтру для коефіцієнта 0,05

Таким чином найкращим коефіцієнтом для використовуваного комплементарного фільтру є  $k=0,1$ . Для цього коефіцієнту перевіримо роботу



фільтру для вхідних сигналів, що були зображені на рис. 3.3 й рис. 3.4. Графік статичної роботи приладу зображено на рис. 3.11.

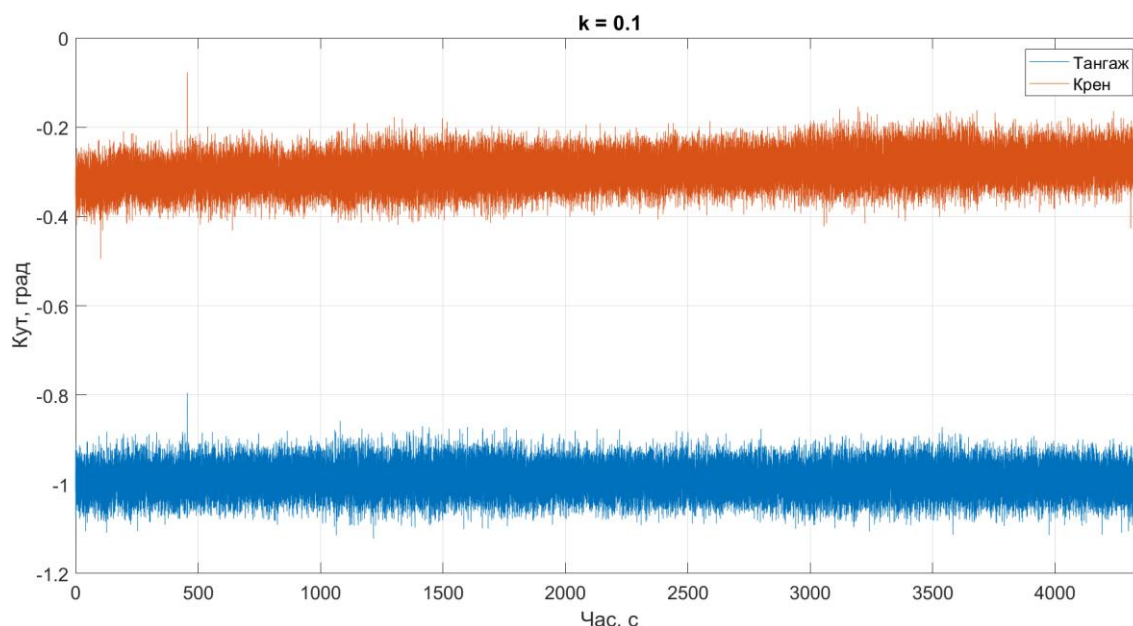


Рисунок 3.11 Статична робота інклінометру з коефіцієнтом комплементарного фільтру  $k=0,1$

Як видно з представленого графіку на рис. 3.11, значення врахованого кута зберігається візуально стабільним протягом тривалого проміжку часу. За час роботи приладу дрейф крену склав 0,0144 град, тангажа – 0,0036 град, швидкість дрейфу крену –  $23,98 \cdot 10^{-6}$  град/с, тангажа –  $5,93 \cdot 10^{-6}$  град/с.

### *Висновки до розділу 3*

В цьому розділі розглянуто вибір коефіцієнту для комплементарного фільтру приладу емпіричним методом та вплив коефіцієнта на роботу приладу. Було розглянуто 6 коефіцієнтів та їх вплив на якість роботи приладу. Для занадто великих коефіцієнтів було справедливим зашумленість вихідних кутів через значний вплив на алгоритм показань акселерометрів. Для занадто малих значень спостерігалась повільність зміни кутів, які значною мірою відставали від заданих для приладу змін. Для даної реалізації найоптимальнішим коефіцієнтом виявився  $k=0,1$ , за якого зберігалась швидкодія відображення та прийнятна точність показань кута. Також за такого коефіцієнту залишається низький дрейф кутів

орієнтації. Так для швидкість дрейфу крену –  $23,98 \cdot 10^{-6}$  град/с, тангажа –  $5,93 \cdot 10^{-6}$  град/с.

## Висновки

В результаті дипломного проекту розроблено інклінометр та його алгоритм роботи. Алгоритм інклінометру базується на використанні МНК для визначення кутового положення приладу відносно горизонту. Оновлення МНК виконується з використанням рівняння Пуассона. Кінематичне рівняння Пуассона інтегрується за допомогою методу лівих прямокутників. Для компенсації накопичення похибки інтегрування використовується комплементарний фільтр. Компенсацією виступають значення кутів визначених за допомогою акселерометрів.

Сам інклінометр має у своєму складі 4 конструктивних блоки: блок чутливих елементів, блок оброблювача інформації, блок живлення та блок відображення. У якості блоку ЧЕ виступає МЕМС датчик MPU6050, який містить у своєму складі трьохосьовий блок ММА та трьохосьовий блок ММГ. Блоком оброблювача інформації виступає апаратна платформа Arduino, а саме відладочна плата Arduino Uno. Блоком живлення виступають два акумулятори типу 18650 з платою контролю заряду. У блок відображення входить дві сервомашини НК15178, кутове положення валів котрих відповідає кутовому положенню приладу.

Програмне середовище Matlab дозволяє створювати програмне забезпечення для вибраної платформи за допомогою двох пакетів: Arduino Support Package for Matlab та Simulink Support Package for Arduino Hardware.

Пакет Arduino Support Package for Matlab дозволяє створювати прошивку для системи лише для роботи з нею в режимі прямого підключення до Matlab. В якості керуючих елементів виступають функції за допомогою яких налаштовують роботу компонентів: створення об'єкту плати, сервомашин, датчика; зняття інформації з датчика; встановлення значення положення валу сервомашин. Алгоритм інклінометру записується у нескінченний цикл для постійного виконання доки користувач не припинить процес, зупинкою виконання коду, розривом з'єднання чи знеструмленням плати.

Пакет Simulink Support Package for Arduino Hardware дозволяє не тільки створення прошивки, для її роботи в режимі прямого підключення до Matlab, а й виконувати завантаження на плату за умови відповідності програми. В якості керуючих елементів виступають блоки, кожний з котрих має відповідну функцію при виконанні програми. Налаштування моделі та параметрів взаємодію з платою Arduino виконується в налаштуваннях моделі. Для зчитування даних з датчика використовується блок «MPU6050 IMU Sensor» в якому задаються необхідні діапазони зчитування даних та полоса пропускання для акселерометрів та гіроскопів, а також їх час опитування. Для взаємодії з сервомашинами використовуються окремі блоки «Standard Servo Write». Основний алгоритм роботи інклінометра, для простоти, записується у блок «MATLAB function» в якому весь запис ведеться кодом Matlab, що дозволяє використовувати той же самий основний код, що і для реалізації через Arduino Support Package for Matlab.

Для коректної роботи приладу необхідно вибрати коефіцієнт комплементарного фільтру таким чином, щоб він забезпечував максимально достовірне відображення кута. Вибір коефіцієнта виконується емпіричним шляхом. Тестування коефіцієнтів виконувалось для двох типів сигналів, які відображають поведження системи для динамічних переміщень та статичної роботи. Серед розглянутих коефіцієнтів вибір здійснений для значення у 0,1. Обраний коефіцієнт зберігає показання при нерухомості системи, має достатню швидкість відображення кутів при зміні кутового положення системи та не дозволяє не сильно реагувати на лінійні прискорення системи. Швидкість дрейфу крену –  $23,98 \cdot 10^{-6}$  град/с, тангажа –  $5,93 \cdot 10^{-6}$  град/с.

## Література

1. С. С. Сысоева, “Тенденции рынка High-end МЭМС-датчиков инерции. Новые уровни характеристик и исполнения,” *Компоненты и технологии*, №. 6, С. 40–46, 2014.
2. Сосков А.Г., Колонтаєвський Ю.П. Промислова електроніка: Підручник. / За ред. А.Г. Соскова. – К.: Каравела, 2015. – 536 с.
3. Peter J. Denning, Ted G. Lewis. Communications of the ACM, January 2017, Vol. 60 No. 1, Pages 54-65
4. Arduino: About Us: веб-сайт. URL: <https://www.arduino.cc/en/Main/AboutUs> (дата звернення 05.06.2020)
5. ЭБС "Университетская библиотека онлайн" читать электронные книги: веб-сайт. URL: <https://biblioclub.ru/index.php?page=dict&termin=708064> (дата звернення 25.12.2019)
6. ЭБС "Университетская библиотека онлайн" читать электронные книги: веб-сайт. URL: <https://biblioclub.ru/index.php?page=dict&termin=750497> (дата звернення 25.12.2019)
7. ЭБС "Университетская библиотека онлайн" читать электронные книги: веб-сайт. URL: <https://biblioclub.ru/index.php?page=dict&termin=1200444> (дата звернення 25.12.2019)
8. КРЕНОМЕТР - это... Что такое КРЕНОМЕТР? : веб-сайт. URL: <https://dic.academic.ru/dic.nsf/sea/4253/%D0%9A%D0%A0%D0%95%D0%9D%D0%9E%D0%9C%D0%95%D0%A2%D0%A0> (дата звернення 25.12.2019)
9. Кренометр КМ веб-сайт. URL: <https://planeta-sos.com/navshstur/krenometr-km/> (дата звернення 25.12.2019)
10. 51.Жидкостной маятниковый переключатель, назначение и устройство. : веб-сайт. URL: <https://studfile.net/preview/3213331/page:11/> (дата звернення 25.12.2019)
11. Строкач Г.Ю, Сапегін О.М. Кутомір на основі акселерометру ADXL-335. *ПОГЛЯД У МАЙБУТНЄ ПРИЛАДОБУДУВАННЯ*: збірник праць XII

Всеукраїнської науково-практичної конференції студентів, аспірантів та молодих вчених, 15-16 травня 2019 р. ↓ К.: ПБФ, КПІ ім. Ігоря Сікорського, Центр учбової літератури. – 2019. – С 46-48

12. Мелешко В.В., Нестеренко О.И. Бесплатформенный инерциальные навигационные системы. Учебное пособие. – Кировоград: ПОЛИМЕД – Сервис, 2011. -172 с.

13. Лазарєв Ю. Ф. Алгоритми безплатформових інерціальних систем орієнтації (прикладні методи синтезу і аналізу точності) [Електронний ресурс] : навч. посіб. для студ. спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології», спеціалізації «Комп'ютерно-інтегровані технології та системи навігації і керування» / Ю. Ф. Лазарєв, О. М. Сапегін ; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 6,8 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2018. – 141 с.

14. MPU-6050-InvenSense-datasheet-21631037.pdf: веб-документ. URL: <https://datasheet.octopart.com/MPU-6050-InvenSense-datasheet-21631037.pdf> (дата звернення 25.12.2019)

15. PS-MPU-9250A-01-v1.1.pdf: веб-документ. URL: <https://invensense.tdk.com/wp-content/uploads/2015/02/PS-MPU-9250A-01-v1.1.pdf> (дата звернення 14.05.2020)

16. iNEMO inertial module: 3D accelerometer, 3D gyroscope, 3D magnetometer: веб-документ. URL: <https://www.st.com/resource/en/datasheet/lsm9ds1.pdf> (дата звернення 14.05.2020)

17. BNO055 Datasheet: веб-документ. URL: <https://www.bosch-sensortec.com/media/boschsensortec/downloads/datasheets/bst-bno055-ds000.pdf> (дата звернення 14.05.2020)

18. MATLAB Support Package for Arduino Hardware Documentation: веб-сайт. URL: <https://www.mathworks.com/help/supportpkg/arduinoio/index.html> (дата звернення 14.05.2020)

19. Simulink Support Package for Arduino Hardware Documentation: веб-сайт. URL: <https://www.mathworks.com/help/supportpkg/arduino/index.html> (дата звернення 14.05.2020)
20. Лазарєв Ю. Ф. Кінематика твердого тіла: навчальний посібник. – К.: НТУУ "КПІ", 2014. – 61 с. [електронний ресурс: <http://kafpson.kpi.ua/arch.htm>]
21. Tariqul Islam, Md.Saiful Islam, Md.Shajid-Ul-Mahmud, Md Hossam-E-Haider. Comparison of Complementary and Kalman Filter Based Data Fusion for Attitude Heading Reference System. AIP Conference Proceedings 1919, 020002 (2017); URL: <https://aip.scitation.org/doi/pdf/10.1063/1.5018520> (дата звернення 30.05.20)
22. Shane Colton. The Balance Filter. Веб-документ URL: <https://docs.google.com/viewer?a=v&pid=sites&srcid=ZGVmYXVsdGRvbWFpbnxteWltdWVzdGltYXRpb25leHBlcmllbmNlfGd4OjY1Yzk3YzhiZmE1N2M4Y2U> (дата звернення 30.05.20)
23. Математичне моделювання на ЕОМ (Частина 2) [Електронний ресурс] : методичні вказівки до виконання комп'ютерних практикумів для студентів напряму підготовки 6.051003 «Приладобудування» / НТУУ «КПІ» ; уклад. Ю. Ф. Лазарєв, Д. О. Півторак, С. Л. Лакоза. – Електронні текстові данні (1 файл: 1,08 Мбайт). – Київ : НТУУ «КПІ», 2015. – 86 с. – Назва з екрана.
24. Arduino Uno Rev3 | Arduino Official Store веб-сайт. URL: <https://store.arduino.cc/arduino-uno-rev3> (дата звернення 05.06.20)
25. HobbyKing <sup>TM</sup> HK15178 Analog Servo 1,4кг / 0.09sec / 10Г веб-сайт. URL: [https://hobbyking.com/ru\\_ru/hobbykingtm-hk15178-analog-servo-1-4kg-0-09sec-10g.html?store=ru\\_ru](https://hobbyking.com/ru_ru/hobbykingtm-hk15178-analog-servo-1-4kg-0-09sec-10g.html?store=ru_ru) (дата звернення 05.06.20)

## Додаток А

Програмний код для виконання у пакеті Arduino Support Package for Matlab.

```
%% setup
clear C0 wx0 k C wx Phi Tet Psi a imu s1 s2
% hold all
%% const value
% start condition
fs=100; %samples/secs
k = 0.1; % compementary filter coef
%% constant value
rtd=180/pi; %radian to degrees
dtr=pi/180; %degrees to radian
%% colibration coef for imu
Rinv=[0.0171343003859391 -0.000244753903015213 -0.000919098353891202;
      -0.000141664004177236 0.0174839032224351 0.000197397119723198;
      0.00118164771642437 -0.000126196715770949 0.0174675806882357;]; % for gyro
Uw0 = [0.0341923523822592; 0.0377819336208767; 0.000765718118321476]; % for gyro
zero bias
Kinv = [-0.0152884517952203,-1.00287331017347,-0.00699593461705786;
        0.992727589585826,-0.0278828935577812,0.0518729811702827;
        -0.0603571328067325,-0.00382622238357238,0.987938224725119]; % for accel
Ua0 =[-0.0494879433558887; 0.102583678830311; 0.409136400873429]; % for accel zero
bias
%% connect arduino, mpu6050
%creating arduino object
a=arduino('COM3','Uno'); %'COM6','Nano3'
%creating servo object
s1=servo(a,'D9');
s2=servo(a,'D10');
%creating mpu object device function
imu=mpu6050(a,'SampleRate',fs,'SamplesPerRead',1,'OutputFormat','matrix'); %
%% start condition
[accelReadings, gyroReadings, timeStamps] = read(imu);
axlx = accelReadings(1,1); % imagine acceleration at x axis
axly = accelReadings(1,2); % imagine acceleration at y axis
axlz = accelReadings(1,3);
g = sqrt(axlx^2+axly^2+axlz^2);
c11 = axlx/g; c12 = axlz/g; c13 = -axly/g;
psi = -atan2(c11,c12);
tet = atan2(c13,sqrt(1-c13^2));
phi = 0;
% start rotation matrix
C0=[ (cos(tet)*cos(phi)) (-cos(tet)*sin(phi)) (sin(tet));
     (cos(psi)*sin(phi)+sin(psi)*cos(phi)*sin(tet)) (cos(psi)*cos(phi)-
sin(psi)*sin(phi)*sin(tet)) (-sin(psi)*cos(tet));
     (sin(psi)*sin(phi)-cos(psi)*cos(phi)*sin(tet))
     (sin(psi)*cos(phi)+cos(psi)*sin(phi)*sin(tet)) (cos(psi)*cos(tet));];
writePosition(s1,ang2serv(atan2(-C0(2,3),C0(3,3))*rtd));
writePosition(s2,ang2serv(asin(C0(1,3))*rtd));
% start skew-symmetric matrix
wx0=[0 0 0;0 0 0;0 0 0];
j=1;
innerTime = 0;
t1 = timeStamps.Second;
t2 = 0;
%% loop
while(true)
%% read data from mpu by device function
[accelReadings, gyroReadings, timeStamps] = read(imu);
```



```

%% time computing
t2 = timeStamps.Second;
if t2 > t1
    h = t2-t1;
else
    h = 60-t1+t2;
end
innerTime = innerTime+h;
%%
Uw = Rinv*(gyroReadings'-Uw0)*dtr;
Ua = Kinv*(accelReadings'-Ua0);
axlx = Ua(1,:); % imagine acceleration at x axis
axly = Ua(2,:); % imagine acceleration at y axis
axlz = Ua(3,:); % imagine acceleration at z axis
omX = Uw(1,:); % angular rate x axis
omY = Uw(2,:); % angular rate y axis
omZ = Uw(3,:); % angular rate z axis
wx=[0 -omZ omY;
    omZ 0 -omX;
    -omY omX 0]; % skew-symmetric matrix
%% rotation matrix by accelerometers
g = sqrt(axlx^2+axly^2+axlz^2);
psi = acos(axlx/g)-pi/2;
tet = acos(axly/g)-pi/2;
phi = acos(axlz/g)-pi/2;
Ca=[(cos(tet)*cos(phi)) (-cos(tet)*sin(phi)) (sin(tet));
    (cos(psi)*sin(phi)+sin(psi)*cos(phi)*sin(tet)) (cos(psi)*cos(phi)-
sin(psi)*sin(phi)*sin(tet)) (-sin(psi)*cos(tet));
    (sin(psi)*sin(phi)-cos(psi)*cos(phi)*sin(tet))
(sin(psi)*cos(phi)+cos(psi)*sin(phi)*sin(tet)) (cos(psi)*cos(tet));];
%% rectangle method
Cw=C0+(C0*wx0)*h;%mnk in rad
%% complementary filter
C=(1-k)*(Cw)+(k*Ca);
%% compute time
Psi=atan2(-C(2,3),C(3,3));
Tet=asin(C(1,3));
%% write data to servo
writePosition(s1,ang2serv(Psi*rtd));%declaring angle to servo
writePosition(s2,ang2serv(Tet*rtd));
%% refresh data
t1 = t2;
j = j+1;
C0 = C;
wx0 = wx;
if mod(j,122) == 0
    imu.release;
end
end

function ats=ang2serv(angle)
% recalculate from angle to servo value
% servo value must be from 0 to 1
%
if angle>0
    ats=0.5*(1+abs(angle/90));
elseif angle<0
    ats=0.5*(1-abs(angle/90));
else
    ats=0.5;
end

```

```

if ats>1
    ats=0.94;
end
if ats<0
    ats=0.06;
end
end

```

## Програмний код для функції «Matlab Function» у пакеті Simulink Support Package for Arduino Hardware.

```

function [Phi,Tet,C,wx] = inclLoop(Ua,Uw,C0,wx0)
%#codegen
%%
k = 0.1; % filter coefficient
h = 0.01; % step
Rinv = [0.0171343003859391 -0.000244753903015213 -0.000919098353891202;
-0.000141664004177236 0.0174839032224351 0.000197397119723198;
0.00118164771642437 -0.000126196715770949 0.0174675806882357]*pi/180;
Uw0 = [0.0341923523822592; 0.0377819336208767; 0.000765718118321476]*pi/180;
Kinv = [-0.0152884517952203,-1.00287331017347,-0.00699593461705786;
0.992727589585826,-0.0278828935577812,0.0518729811702827;
-0.0603571328067325,-0.00382622238357238,0.987938224725119];
Ua0 = [-0.0494879433558887; 0.102583678830311; 0.409136400873429];
%%
Uw = Rinv*(Uw-Uw0); % calibrate gyroscope
Ua = Kinv*(Ua-Ua0); % calibrate accelerometer
omX = Uw(1,:); % angular rate x axis
omY = Uw(2,:); % angular rate y axis
omZ = Uw(3,:); % angular rate z axis
axlx = Ua(1,:); % imagine acceleration at x axis
axly = Ua(2,:); % imagine acceleration at y axis
axlz = Ua(3,:); % imagine acceleration at z axis
wx = [0 -omZ omY;
omZ 0 -omX;
-omY omX 0]; % skew-symmetric matrix
%% accel rotating matrix
g = sqrt(axlx^2+axly^2+axlz^2);
psi = acos(axlx/g)-pi/2;
tet = acos(axly/g)-pi/2;
phi = acos(axlz/g)-pi/2;
Ca=[ (cos(tet)*cos(phi)) (-cos(tet)*sin(phi)) (sin(tet));
(cos(psi)*sin(phi)+sin(psi)*cos(phi)*sin(tet)) (cos(psi)*cos(phi)-
sin(psi)*sin(phi)*sin(tet)) (-sin(psi)*cos(tet));
(sin(psi)*sin(phi)-cos(psi)*cos(phi)*sin(tet))
(sin(psi)*cos(phi)+cos(psi)*sin(phi)*sin(tet)) (cos(psi)*cos(tet));];
%% numerical integration
Cw=C0+(C0*wx0)*h;
%% filter equation
C=(1-k)*(Cw)+(k*Ca);
%% angle computing
Phi=atan2(-C(2,3),C(3,3));
Tet=asin(C(1,3));
end

```

Запис початкових умов в блоці «Unit Delay» для МНК.

```

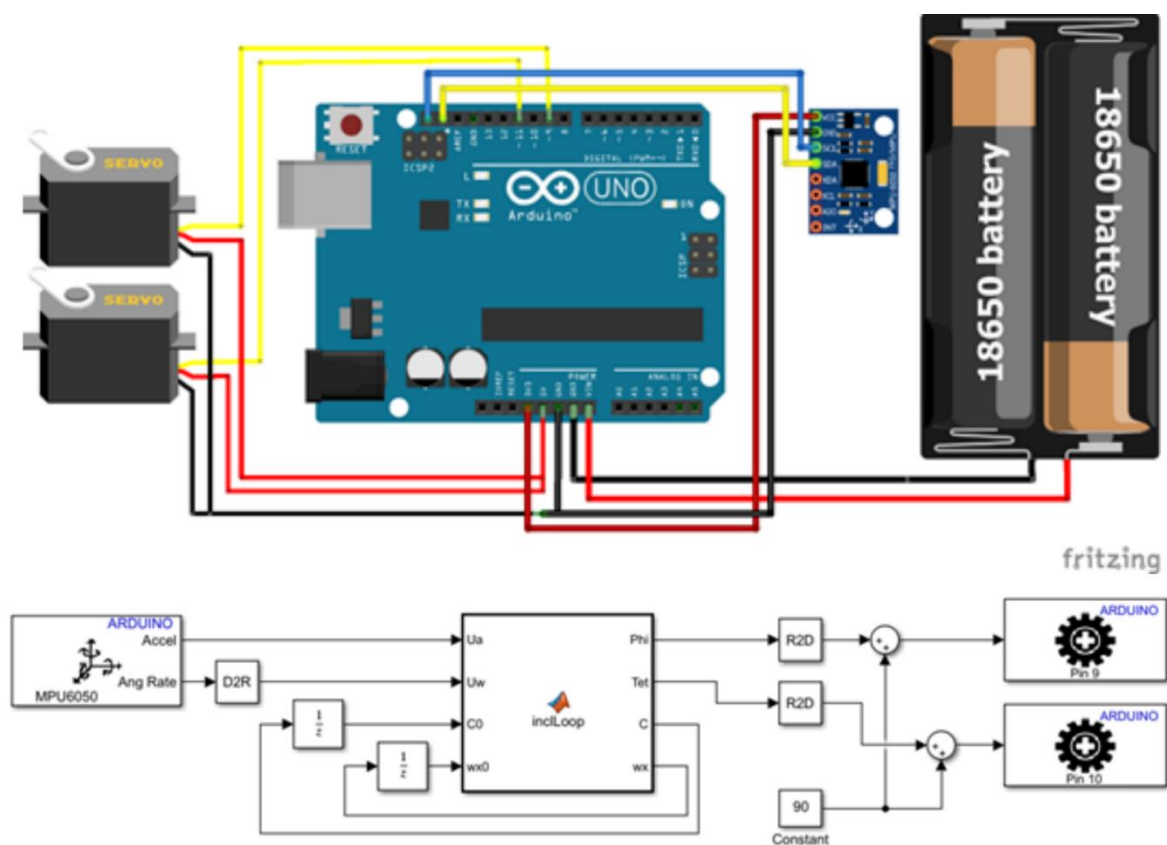
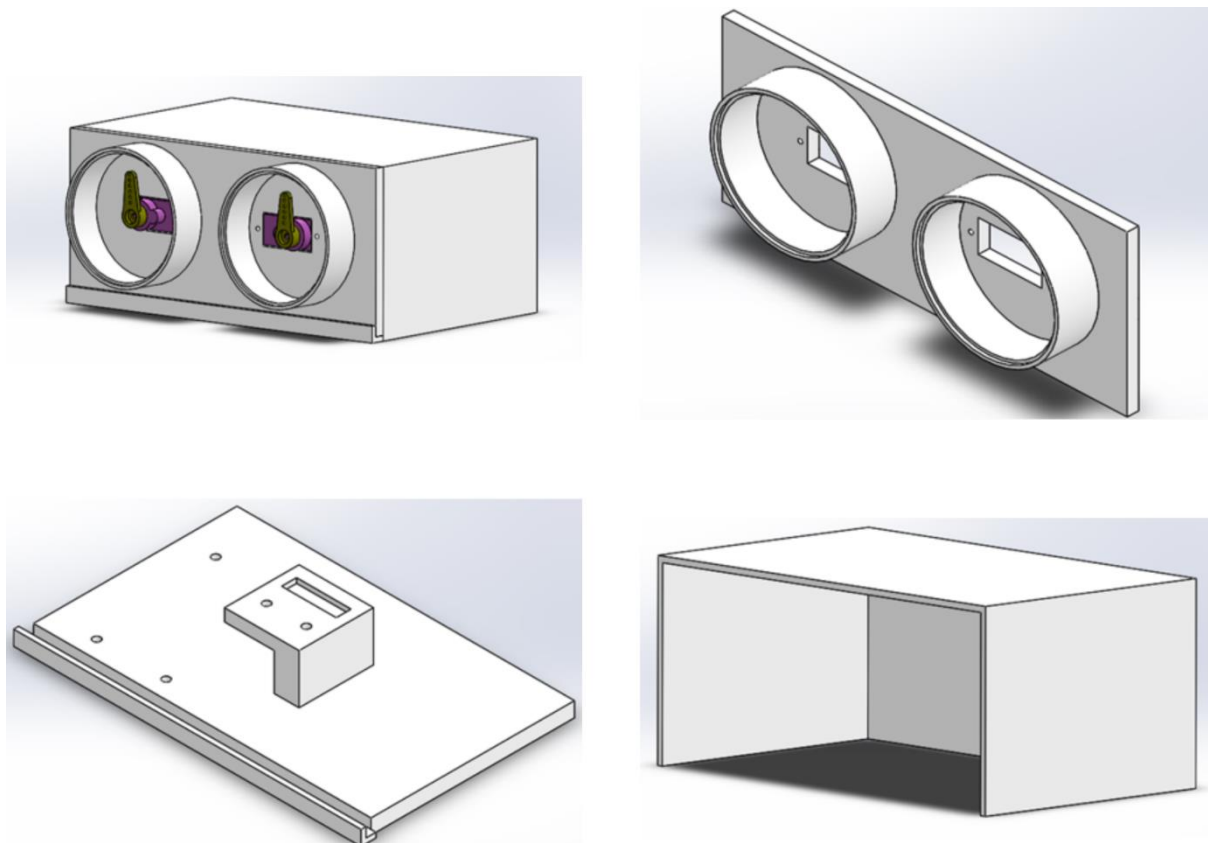
[ (cos (atan2 (-Ua (2,1)/9.81,sqrt (1-(-Ua (2,1)/9.81)^2))) *cos (0)) (-cos (atan2 (-
Ua (2,1)/9.81,sqrt (1-(-Ua (2,1)/9.81)^2))) *sin (0)) (sin (atan2 (-Ua (2,1)/9.81,sqrt (1-(-
Ua (2,1)/9.81)^2)))));
(cos ( -atan2 (Ua (1,1)/9.81,Ua (3,1)/9.81)) *sin (0)+sin ( -
atan2 (Ua (1,1)/9.81,Ua (3,1)/9.81)) *cos (0) *sin (atan2 (-Ua (2,1)/9.81,sqrt (1-(-
Ua (2,1)/9.81)^2)))) (cos ( -atan2 (Ua (1,1)/9.81,Ua (3,1)/9.81)) *cos (0)-sin ( -
atan2 (Ua (1,1)/9.81,Ua (3,1)/9.81)) *sin (0) *sin (atan2 (-Ua (2,1)/9.81,sqrt (1-(-
Ua (2,1)/9.81)^2)))) (-sin ( -atan2 (Ua (1,1)/9.81,Ua (3,1)/9.81)) *cos (atan2 (-
Ua (2,1)/9.81,sqrt (1-(-Ua (2,1)/9.81)^2)))));
(sin ( -atan2 (Ua (1,1)/9.81,Ua (3,1)/9.81)) *sin (0)-cos ( -
atan2 (Ua (1,1)/9.81,Ua (3,1)/9.81)) *cos (0) *sin (atan2 (-Ua (2,1)/9.81,sqrt (1-(-
Ua (2,1)/9.81)^2)))) (sin ( -atan2 (Ua (1,1)/9.81,Ua (3,1)/9.81)) *cos (0)+cos ( -
atan2 (Ua (1,1)/9.81,Ua (3,1)/9.81)) *sin (0) *sin (atan2 (-Ua (2,1)/9.81,sqrt (1-(-
Ua (2,1)/9.81)^2)))) (cos ( -atan2 (Ua (1,1)/9.81,Ua (3,1)/9.81)) *cos (atan2 (-
Ua (2,1)/9.81,sqrt (1-(-Ua (2,1)/9.81)^2)))));]

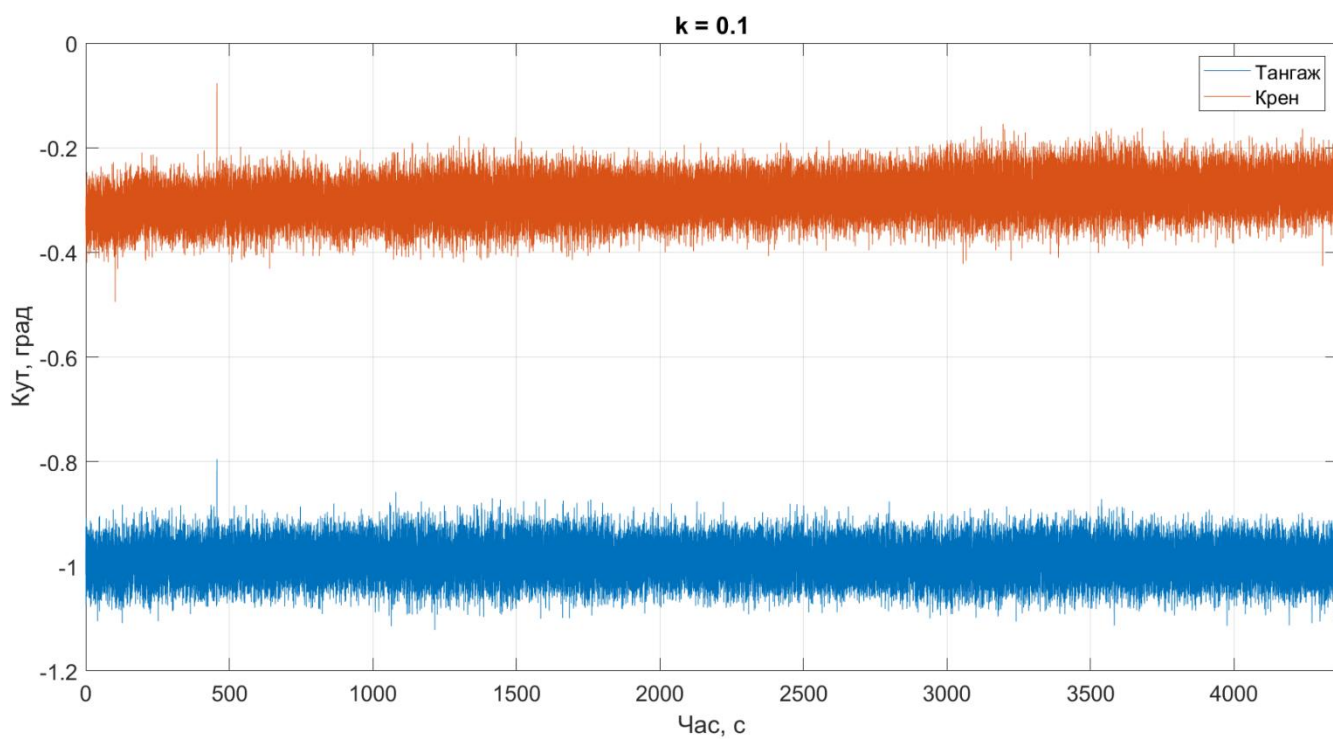
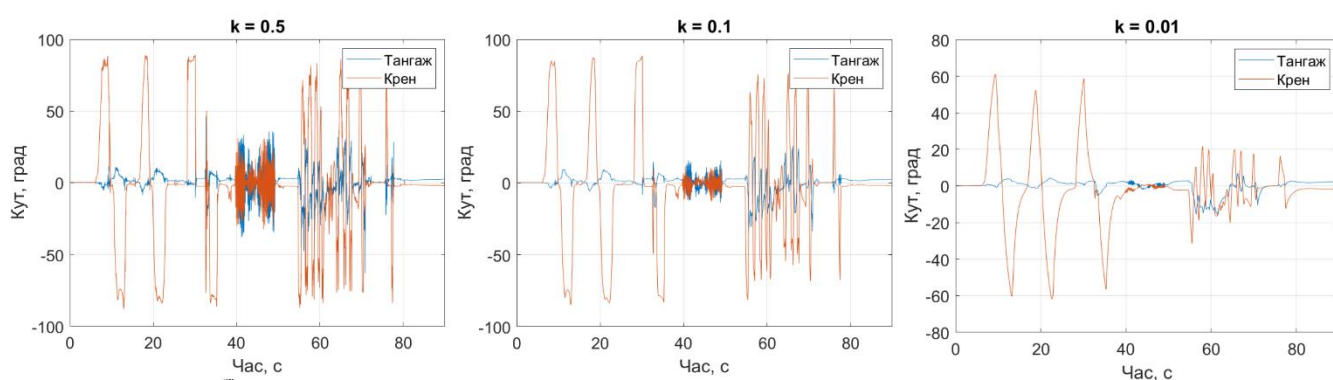
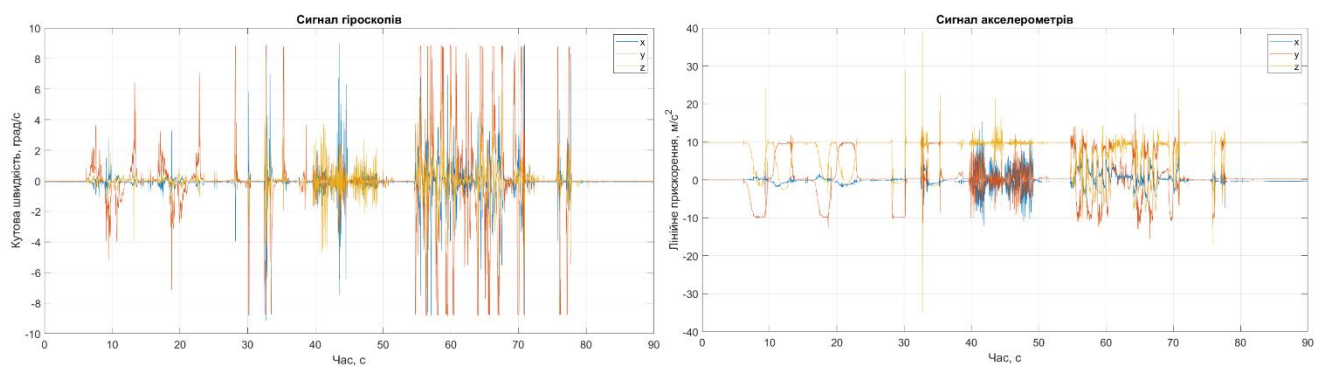
```

**Запис початкових умов в блоці «Unit Delay» для кососиметричної матриці .**



```
[0 0 0; 0 0 0; 0 0 0]
```

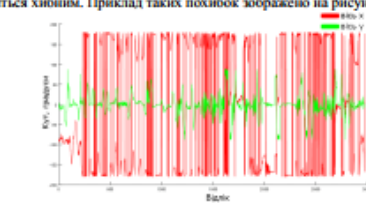
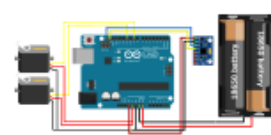
## Додаток Б





## Додаток В

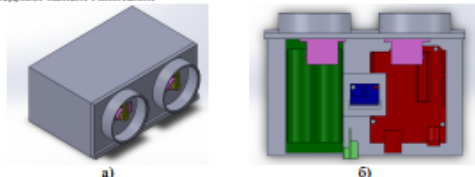
 <p><b>МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ</b>  <b>НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ</b>  <b>«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ</b>  <b>імені ІГОРЯ СІКОРСЬКОГО»</b>  <b>ПРИЛАДОБУДІВНИЙ ФАКУЛЬТЕТ</b></p> <p><i>XIII Науково-практична конференція студентів, аспірантів та молодих вчених</i></p> <p><b>"ПОГЛЯД У МАЙБУТНЄ ПРИЛАДОБУДУВАННЯ"</b></p> <p>13-14 травня 2020 р. м. Київ, Україна</p> <p><b>Збірник праць конференції</b></p>  <p>КИЇВ 2020</p>	<p>XIII Всеукраїнська науково-практична конференція студентів, аспірантів та молодих вчених «Погляд у майбутнє приладобудування», 13-14 травня 2020 року, КПІ ім. Ігоря Сікорського, м. Київ, Україна</p> <p><i>Г.Ю. Строчак, студент гр. ПГ-61, асистент Савенів О.М.</i>  <b>КУТОМІР НА ОСНОВІ МІКРОМЕХАНІЧНОГО ІНЕРЦІАЛЬНОГО ВІМІРЮВАЛЬНОГО МОДУЛЯ</b>..... 67</p> <p><b>СЕКЦІЯ 2. ОПТИЧНІ ТА ОПТИКО-ЕЛЕКТРОННІ ПРИЛАДИ І СИСТЕМИ</b></p> <p><i>І.В. Герасим, студент гр. ПМ-61</i>  <b>СТАБІЛІЗАЦІЯ ОПТИЧНИХ ПРИЛАДІВ</b>..... 72</p> <p><i>О.С. Гудим, аспірант гр. ПМ-91, д.т.н., професор Барановської В.М.</i>  <b>ОПТИЧНІ МІКРОСКОПИ ЗІ СТРУКТУРОВАНИМ ОСВІТЛЕННЯМ</b>..... 76</p> <p><i>Є.О. Лазарев, студент гр. ПМ-61</i>  <b>ЛАЗЕРНІ РІВНІ</b>..... 79</p> <p><b>СЕКЦІЯ 3. КОМП'ЮТЕРНО-ІНТЕГРОВАНІ ТЕХНОЛОГІЇ ВИРОБНИЦТВА ПРИЛАДІВ</b></p> <p><i>О. Савалова, студент гр. ПМ-61, д.т.н., асистент Прокопчук М.Д.</i>  <b>MODELING OF TITANIUM ALLOYS PROCESSING PARAMETERS</b>..... 84</p> <p><i>О. Савалова, студент гр. ПМ-61, д.т.н., асистент Прокопчук М.Д.</i>  <b>MODELING OF PARAMETERS OF THE MILLING OF PARTS FROM HEAT-RESISTANT STEEL</b>..... 88</p> <p><i>Бельман О.І. студент групи ПМ-61, д.т.н., к.т.н. Степанюк В.В.</i>  <b>АВТОМАТИЗОВАНА СИСТЕМА СОРТУВАННЯ ВІДХОДІВ НА БАЗІ ВІЗУАЛЬНОЇ СПЕКТРОМЕТРИЇ</b>..... 92</p> <p><i>О.С. Гавриш, студент гр. ПМ-91</i>  <b>ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ РОБОТА МАНІПУЛЯТОРА</b>..... 95</p> <p><i>О.В. Карпенко, студент гр. ПМ-91</i>  <b>СИСТЕМА ОПЕРАТИВНОГО МОНИТОРИНГУ СТАНУ ВІПРОДВИГУНА</b>..... 98</p> <p><i>К.В. Крижанівський, студент гр. ПМ-61, професор Герасимук М.Д.</i>  <b>ВИРОБНИЦТВО ЗАГОТОВОК (СЛЯБІВ) В МЕТАЛУРГІЇ</b>..... 101</p> <p><i>В.В. Марченко, студент гр. ПМ-61, к.т.н., ст. викладач Матвійчук С.М.</i>  <b>МЕТОДИ БЕЗДРОТОВОЇ ПЕРЕДАЧІ ІНФОРМАЦІЇ В ПРИЛАДОБУДУВАННІ</b>..... 105</p> <p><i>І.В. Мостовий, студент гр. ПМ-91, к.т.н., д.т.н. Степанюк В.В.</i>  <b>АНАЛІЗ МЕТОДІВ ТОПОЛОГІЧНОЇ ОПТИМІЗАЦІЇ ПРИ ПРОЕКТУВАННІ ЕЛЕМЕНТІВ ПРИЛАДІВ</b>..... 109</p> <p><i>В.В. Лопатинський, студент гр. ПМ-61, асистент Волонин О.В.</i>  <b>МЕТОДИКА ВИЗНАЧЕННЯ ОПТИМАЛЬНИХ ПАРАМЕТРІВ ДЕТАЛЕЙ ТА ЇХ ВИГОТОВЛЕННЯ</b>..... 112</p> <p><i>Д.М. Прокопчук, студент гр. ПМ-61</i>  <b>СИСТЕМА УПРАВЛІННЯ МАНІПУЛЯТОРАМИ</b>..... 116</p> <p><i>Д.М. Прокопчук, студент гр. ПМ-61</i>  <b>РОЗРОБКА ЛАБОРАТОРНОГО МАНІПУЛЯТОРА</b>..... 119</p> <p>6</p>
--	---

<p>XIII Всеукраїнська науково-практична конференція студентів, аспірантів та молодих вчених «Погляд у майбутнє приладобудування», 13-14 травня 2020 року, КПІ ім. Ігоря Сікорського, м. Київ, Україна</p> <p>УДК 631.451</p> <p><i>Г.Ю. Строчак, студент гр. ПГ-61, асистент Савенів О.М.</i>  <b>КУТОМІР НА ОСНОВІ МІКРОМЕХАНІЧНОГО ІНЕРЦІАЛЬНОГО ВІМІРЮВАЛЬНОГО МОДУЛЯ</b></p> <p><i>Анотація.</i> В даній статті розглядається новий варіант створення кутоміра на основі мікромеханічного датчика, що включає в себе блок акселерометрів та гіроскопів на основі плати Arduino у програмному середовищі Matlab.</p> <p><i>Ключові слова:</i> кутомір, мікромеханічний датчик, Arduino, Matlab.</p> <p><b>ВСТУП</b></p> <p>Arduino являє собою електронну платформу, головною ідеєю якої є забезпечення легкості при використанні апаратного та програмного забезпечення. Тобто для використання цієї платформи не треба дуже глибоко вивчати принципи роботи мікроконтролерів та периферії для них, відкритість початкового коду (open source) дає можливість модифікації та створення різноманітних модифікацій та поліпшень. Так завдяки цьому існує велика кількість різноманітних бібліотек створених користувачами для подальшої оптимізації та покращення роботи користувачів. Через це стало можливим створення додатку для середовища Matlab – Matlab Support Package for Arduino Hardware, що дозволяє виконувати взаємодію з платою Arduino. В сам додаток вбудовано бібліотеки для роботи з послідовними інтерфейсами ("I2C", "SPI"), серво-машинами ("Servo"), а також існують спеціальні функції для спрощення роботи з найпопулярнішими інерційними вимірювальними модулями (MPU6050, MPU9250, LSM9DS1) робота з якими забезпечується функціями, що мають назву модульм.[1]</p> <p><b>ПОСТАНОВКА ЗАДАЧІ</b></p> <p>В минулій статті авторів було успішно продемонстровано створення кутоміру на основі акселерометрів.[2] Проте, такий прилад мав один суттєвий недолік – при русі об'єкта, на якому встановлено прилад, акселерометри починають реагувати на прискорення об'єкту і виміряне значення кутів становиться хибним. Приклад таких похибок зображено на рисунку 1.</p>  <p>Рисунок 1. Покази кутоміру на основі акселерометрів на рухому об'єкті</p> <p>67</p>	<p><b>СЕКЦІЯ №1 — КОМП'ЮТЕРНО-ІНТЕГРОВАНІ ТЕХНОЛОГІЇ ТА СИСТЕМИ НАВІГАЦІЇ І КЕРУВАННЯ</b></p> <p>При тестуванні на рухомому транспорті імітувались нахили, подібні до тих, які може здійснювати автомобіль на бездоріжжі, шляхом невеликих нахилів самого модулю. Вісь X була направлена по колінарно до напрямку руху транспорту, вісь Y – перпендикулярно до X. Як можна побачити з рисунку 1, вихідний кут осі X (червоний) зовсім не відображає значення нахилу. Через це становиться неможливим використання описуваного в статті приладу на рухомих основах. Тож для уникнення такої проблеми необхідно додатково використовувати гіроскопи і разом з акселерометрами створити безплатформну інерціальну навігаційну систему. А для мобільності системи створити для неї устаткування, що забезпечить автономність приладу – корпус, елементи живлення, показачки.</p> <p><b>КОНСТРУКЦІЯ КУТОМІРУ</b></p> <p>Для реалізації кутоміра вибрано такі складові:</p> <ul style="list-style-type: none"> <li>• вимірювальний модуль MPU6050;</li> <li>• плата Arduino UNO;</li> <li>• серво-машини HK15178</li> <li>• акумулятори NCR18650B з кейсом та контролером заряду.</li> </ul> <p>Підключення компонентів один до одного відбувається за схемою зображеною на рисунку 2. Для MPU6050 рекомендована напруга живлення 3,3 В, тому його на нього живлення подається з виходу на платі, а контакти для I2C SDA SCL під'єднані до контактів SDA SCL на платі, також можливе під'єднання до контактів A4 та A5 відповідно. Для серво-машин необхідна напруга живлення в 5 В, яке подається з плати, контакти керування під'єднані до контактів із функцією PWM – 9 та 10. Живлення плати з акумуляторів під'єднано до контакту Vin та GND. Контакти контролера заряду та акумуляторного кейсу з'єднані відповідно до їх полярності.</p>  <p>Рисунок 2. Схема підключення компонентів інклінометру до плати Arduino UNO</p> <p>Для зручного використання приладу розроблено корпус приладу. Корпус складається з трьох окремих деталей: основи, передньої панелі. На основі розташовуються та закріплюються елементи приладу. Для фіксації плати та кейсу акумуляторів передбачено отвори для гвинтів. Посередній основі утворено Г-образну стійку для кріплення вимірювального модуля на гвинтах. Також для уникнення скручення встановлення плати, передбачено жолоб для полусфер приною знизу плати модуля. Для кращої фіксації передньої панелі корпусу створено жолоб попереду основи. Не передній панелі зроблено отвори для встановлення та закріплення серво-машин Самі серво-машини оточені кілками що захищають показачки від зовнішніх пошкоджень та служать кріпленням для циферблату. Кріпляться до корпусу за допомогою гвинтів.</p> <p>68</p>
--	---



XII Всеукраїнська науково-практична конференція студентів, аспірантів та молодих вчених «Погляд у майбутнє: приладобудування», 15-16 травня 2020 року, КПІ ім. Ігоря Сікорського, м. Київ, Україна

Кришка корпусу накриває основу та повторює силует передньої панелі. Задю передавачем виріз для доступу до контролеру заряду. Кріпляться до основи та передньої панелі гвинтами.



Рисunek 3 Трьох видів модель приладу: а) зовнішній вигляд приладу; б) розташування елементів в приладі

#### КАЛІБРУВАННЯ СЕНСОРА

Для коректної роботи та точності показань приладу необхідно провести калібрування сенсора. Оскільки сенсор містить в собі блок акселерометрів та гіроскопів провести їх можна за допомогою методів тестових поворотів. [3] Даний метод описується в навчальному посібнику [3]. Калібрування акселерометрів виконується аналогічно до того, що виконувалося в попередній роботі [2].

Калібрування гіроскопів виконується обертанням датчика на поворотному столі з відомою кутовою швидкістю. Для підвищення точності калібрування коефіцієнтів бажано записати показання гіроскопів на декількох кутових швидкостях. [3]

Калібровані матриці для блоку акселерометрів мають вигляд:

$$A = K^{-1}(U_a - U_{a0}) \quad (1)$$

а для блоку гіроскопів:

$$W = R^{-1}(U_g - U_{g0}) \quad (2)$$

де  $K$  - матриця масштабних коефіцієнтів;  $U_a$  - матриця сигналів;  $U_{a0}$  - матриця зміщення нулів для блоку акселерометрів і  $R$  - матриця масштабних коефіцієнтів;  $U_g$  - матриця сигналів;  $U_{g0}$  - матриця зміщення нулів для блоку гіроскопів. Для використаного датчика їх значення мають вигляд:

$$K = \begin{bmatrix} -0.02 & 1.01 & -0.05 \\ -0.99 & -0.02 & -0.01 \\ -0.01 & 0.06 & 1.01 \end{bmatrix}; \quad U_{a0} = \begin{bmatrix} -0.05 \\ 0.11 \\ 0.41 \end{bmatrix} \quad (3)$$

$$R = \begin{bmatrix} 58.159 & 0.836 & 3.051 \\ 0.516 & 57.198 & -0.619 \\ -3.931 & 0.357 & 57.038 \end{bmatrix}; \quad U_{g0} = \begin{bmatrix} -0.034 \\ 0.034 \\ 0.001 \end{bmatrix} \quad (4)$$

69

секція №1 — КОМП'ЮТЕРНО-ІНТЕГРОВАНІ ТЕХНОЛОГІЇ ТА СИСТЕМИ НАВИГАЦІЇ І КЕРУВАННЯ

#### АЛГОРИТМ КУТОМІРА

У програмному середовищі Matlab створюється об'єкт для роботи з платою Arduino, наприклад для Arduino Uno, що підключається до третього комп'ютерного порту ініціалізація має вигляд  $\text{a}=\text{arduino}(\text{'COM3','Uno'})$ . Надалі створюється об'єкт датчика  $\text{imu}=\text{imu8050}(\text{a})$ , а також об'єкти для серво-машини  $\text{c1}=\text{servo}(\text{a,'D9'})$ ,  $\text{c2}=\text{servo}(\text{a,'D10'})$ . [1]

Зчитуючи кутову швидкість з датчика використовуючи функцію  $\text{read}(\text{imu})$ , можемо визначити матрицю напрямних косинусів із яких потім вирахувати кути відхилення самого кутоміру. Для цього сформувавши початкову матрицю напрямних косинусів використовуючи кути, обчислені з показань акселерометрів за аналогією до того, як це було зроблено в роботі [2], та матрицю проєкцій абсолютної кутової швидкості вимірюваних гіроскопів утворюємо рівняння орієнтації Пуассона. Для обчислення змінення матриці напрямних косинусів рівняння Пуассона інтегрують, наприклад методом прямокутників. З отриманої матриці напрямних косинусів вирахуємо кути, які і показують відхилення. Вирахувані кути передаються на серво-машини, які відображають кути для зображення вихідного кута на шкалі приладу.

#### ВИСНОВОК

Для використання кутоміру на рухомих об'єктах необхідно перейти від використання лише акселерометрів при визначенні кута нахилу об'єкту відносно площини горизонту. Для цього до приладу необхідно додати гіроскопи. Використовуючи акселерометри при початковій виставці можна визначити матрицю напрямних косинусів і надалі використовуючи рівняння орієнтації Пуассона та інтегрування методом прямокутників здійснювати обчислення кутів відхилення кутоміру за відображення яких відповідають серво-машини. Задля забезпечення автономності роботи приладу компоненти приладу оснащуються акумулятором та корпусом.

#### СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] MATLAB Support Package for Arduino Hardware Documentation - Режим доступу: <https://www.mathworks.com/help/supportpkg/arduinoofindex.html>
- [2] Строчак Г.Ю., Савегін О.М. Кутомір на основі акселерометру ADXL-335. ПОГЛЯД У МАЙБУТНЄ: ПРИЛАДОБУДУВАННЯ: збірник праць XII Всеукраїнської науково-практичної конференції студентів, аспірантів та молодих вчених. 15-16 травня 2019 р. — К.: ПБФ, КПІ ім. Ігоря Сікорського, Центр учбової літератури. — 2019. — С 46-48
- [3] Мелешко В.В. Бесплатформенные инерциальные навигационные системы: [Учебное пособие] / В.В. Мелешко, О.И. Нестеренко. — Кировград: Политед-Сервис, 2011. — 172с.

Наук. Кепішник — асистент Савегін О.М.

70

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНИЙ АвіАЦІЙНИЙ УНІВЕРСИТЕТ  
ІНЖЕНЕРНА АКАДЕМІЯ УКРАЇНИ  
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ВОДНОГО ГОСПОДАРСТВА  
ТА ПРИРОДОКОРИСТУВАННЯ  
ЧЕРКАСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ІМЕНІ БОГДАНА ХМЕЛЬНИЦЬКОГО  
WROCLAW UNIVERSITY OF SCIENCE AND TECHNOLOGY



Wroclaw University  
of Science and Technology

### ІНТЕГРОВАНІ ІНТЕЛЕКТУАЛЬНІ РОБОТОТЕХНІЧНІ КОМПЛЕКСИ (ІРТК-2020)

ТРИНАДЦЯТА МІЖНАРОДНА НАУКОВО-ПРАКТИЧНА  
КОНФЕРЕНЦІЯ

19-20 травня 2020 р.  
Київ, Україна

ЗБІРКА ТЕЗ

Київ  
2020

- Андрєєва О.В. — Вплив інтелектуальних рішень на рівні небезпечного випромінювання у цифровому будинку 217
- Гарашенко О.В., Гарашенко В.І. — Метод контролю процесу магнітного осадження феромагнітних домішок водних середовищ 219
- Граф М.С. — Побудова алгоритму траєкторії руху безпілотного повітряного судна з ітратою висоти 222
- Данченко Я.В. — Побудова математичної моделі контуру регулювання «рішень — концентрація» при очищенні газового конденсату виробництва аміаку 225
- Данченко Я.В., Тарас Б.І., Лещук Д.О. — Система автоматичного регулювання співвідношення газ-повітря у котлоагрегатах водокерівного типу по його кдд 227
- Егоров С.В., Шкваринська Т.Ю. — Збільшення пропускної здатності каналу передачі інформації методом багатопотокового передавання даних 230
- Клепач М.М. — Перехідні характеристики системи управління маніпулятором паралельної структури для координатно-вимірних машин 232
- Коваленко О.О., Васильченко В.Ю., Скоромний В.І. — Розрахунок деталей на міцність та тепловий аналіз методами кінцевих елементів в MATLAB 235
- Коваленко О.О., Крейда А.М., Шарпа В.М. — Оптимізація параметрів технічних пристроїв у складі САПР 238
- Ковальчук В.В., Мамука К.В. — Властивості керметних плівок атомарно-кластерної дисперсності 242
- Ковальчук В.В., Сморж М.В. — Пристрій з нанокластерною підсистемою 244
- Лобченко В.В. — Визначення інтенсивності потоку в комп'ютерних системах із неоднорідним трафіком 246
- Малащенко В.М., Стрілець О.Р., Андрушків В.І., Стрілець В.М. — Обґрунтування будови та принципу роботи муфти флацево-пальцевої пружної запобіжної 248
- Паздрій О.Я. — Підвищення ефективності дворівневої обробки інформації для багатокласової системи діагностики складної обертової системи 251
- Приз В.А. — Узагальнена структурна схема верстата з числовим програмним керуванням 253
- Стрілець О.Р. — Застосування 3-d моделювання для оптимізації розмірів при проектуванні зубчастого диференціала з замкнутою гідросистемою при керуванні швидкістю через водило 255
- Строчак Г.Ю., Савегін О.М. — Фільтр маджика в алгоритмі інклінометра 258
- Хайдуков В.В. — Применение прикладных программных пакетов MATLAB и Comsol Multiphysics при исследовании волновых процессов в твердых телах 260
- Яворський І.М., Юзефович Р.М., Курапов П.Р. — Аналіз кореляційних властивостей перетворення гілберта періодично нестационарного випадкового сигналу 263

11

УДК 631.451

## ФІЛЬТР МАДЖВІКА В АЛГОРИТМІ ІНКЛІНОМЕТРА

Строчак Г.Ю., студент, НТУ України «КПІ ім. Ігоря Сікорського»,  
gr.strochak@gmail.com  
Saperin O.M., асистент, НТУ України «КПІ ім. Ігоря Сікорського»,  
saperin\_o@ukr.net

Поява такої електронної платформи як Arduino, основними ідеями якої є простота при використанні апаратного та програмного забезпечення та відкритість початкового коду, дозволило її користувачам створити велику кількість модифікацій та поліпшень для роботи з платформою. Одним з результатів таких поліпшень є додаток до програмного середовища Matlab – «Matlab Support Package for Arduino Hardware», що дозволяє програмувати плати Arduino безпосередньо в самому середовищі. В додаток вбудовані бібліотеки для взаємодії зі зовнішньою периферією через інтерфейси «I2C» й «SPI», а також для керування сервомашинами бібліотека «Servo».[1]

У статті авторів (2) було успішно створено інклінометр в основі алгоритму роботи котрого головними чутливими елементами нахилу приладу виступали акселерометр. Однак, використання в алгоритмі інклінометра чутливих елементів у вигляді акселерометрів має один суттєвий недолік – при переміщенні приладу разом з об'єктом, на якому він встановлений, акселерометри зчитують прискорення його руху і виміряне значення кутів становиться хибним. Так, якщо вісь X направити по ходу руху об'єкта, а вісь Y, відповідно, перпендикулярно до осі X і виміряти невеликі нахили при русі об'єкта, то вихідне значення інклінометру матиме вигляд графіку на рис. 1.

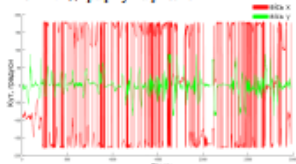


Рис. 1. Покази кутів нахилу на основі акселерометрів на рухомий об'єкт

Як видно з графіка, обчислюваний за алгоритмом кут осі X (червоний) має хаотичний вигляд, що не відповідає заданим значенням нахилу через вплив прискорення об'єкта на чутливі елементи. Саме через це унеможливилось використання лише акселерометрів для роботи рухомих основ. Для вирішення проблеми необхідно разом з акселерометрами використовувати гіроскопи.

В якості алгоритму роботи для системи можна використовувати фільтр, що був розроблений Себастьяном Маджвіком і носить назву свого автора. Сам фільтр виступає альтернативою для фільтра Калмана. Так фільтр оснований на

258

ПРТК-2020

процесі Калмана потребує великі обчислювальні потужності для свого впровадження. Розроблений же Маджвіком фільтр знімає проблеми точності та налаштування параметрів фільтра, що базуються на підходах Калмана і аналізує такі параметри: два параметри для регулювання (одні для систем без магнітометрів), які визначаються із спостережуваних параметрів системи; аналітичні обчислення та оптимізація методом градієнтного спуску, які дають підвищення точності при малих частотах дискретизації; компенсація зміщення нуля та магнітних скривлень у реальному часі. Також фільтр Маджвіка передбачає значне скорочення обчислювального навантаження й дозволяє фільтру оптимально підсилити джерело даних базуючись на спостережуваних характеристиках системи.[3]

Програмний код фільтра Маджвіка був викладений його автором в додатках роботи (3), а також доступний у відкритому вигляді на мовах C, C# та Matlab. Тому можна з легкістю використовувати даний фільтр для роботи з інерціальними вимірювальними модулями (IBM, англ. – Inertial Measurement Unit) чи інерціальними навігаційними системами (ІНС, англ. AHRS – Attitude and Heading Reference Systems) у середовищі Matlab. Так створивши об'єкти для взаємодії з платою Arduino та IBM (ІНС) можна зчитувати з нього значення акселерометрів та гіроскопів (для ІНС додатково – магнітометрів) з постійним, відомим кроком опитування й передавати до об'єкта фільтра значення датчиків. Надалі «втягувати» з об'єкта фільтра кватерніон стану та з нього виходувати кути Ейлера і далі передавати для відображення на графіках, циферблатах тощо. Для IBM, наприклад, кінцеві значення мають бути в розмірності рад/с для гіроскопів та м/с<sup>2</sup> для акселерометрів і тому, для перекладу з цифрового коду, в пакеті «Matlab Support Package for Arduino Hardware» для найбільш поширених датчиків (MPU6050, MPU9250, LSM9DS1) вже вбудовані алгоритми перерахунку на град/с та м/с<sup>2</sup> для них. В сам об'єкт фільтра передається частота опитування IBM та коефіцієнт підсилення фільтра.

Висновок: для використання інклінометру на рухомій основі краще враховувати значення його нахилів за допомогою фільтра Маджвіка, який є простим для налаштування та використання та не потребує великої кількості обчислювальної потужності.

## Література

1. MATLAB Support Package for Arduino Hardware Documentation – Режим доступу: <https://www.mathworks.com/help/supportpkg/arduino/index.html>
2. Строчак Г.Ю., Сперін О.М. Кутимір на основі акселерометру ADXL-335. ПОГЛЯД У МАЙБУТНЄ ПРИЛАДОБУДУВАННЯ: збірник праць XII Всеукраїнської науково-практичної конференції студентів, аспірантів та молодих вчених, 15-16 травня 2019 р. – К.: ПФФ, КПІ ім. Ігоря Сікорського, Центр університетської літератури, – 2019. – С 46-48
3. Sebastian O.H. Madgwick. An efficient orientation filter for inertial and/or magnetic sensor arrays. URL: [https://www.x-io.co.uk/res/doc/madgwick\\_inertial\\_report.pdf](https://www.x-io.co.uk/res/doc/madgwick_inertial_report.pdf)

ПРТК-2020

259



## ПРОБЛЕМИ ТА ПЕРСПЕКТИВИ РЕАЛІЗАЦІЇ ТА ВПРОВАДЖЕННЯ МІЖДИСЦИПЛІНАРНИХ НАУКОВИХ ДОСЯГНЕНЬ

12 ЧЕРВНЯ 2020 РІК

М. КИЇВ, УКРАЇНА

1 ТОМ



DOI 10.36074/12.06.2020.v1  
ISBN 978-617-7171-75-0

Проблеми та перспективи реалізації та впровадження міждисциплінарних наукових досягнень + Том 1

РОЛЬ ФІНАНСОВОЇ СТРАТЕГІЇ В ЗАБЕЗПЕЧЕННІ РОЗВИТКУ ПІДПРИЄМСТВА Фесюк Н.В. ....	38
СУТНІСТЬ ТА ЧИННИКИ ФОРМУВАННЯ ПОТЕНЦІАЛУ ЕКОНОМІЧНОГО РОЗВИТКУ ПІДПРИЄМСТВ Ємельянов О.Ю. ....	38
ТИПИ КОРПОРАТИВНОЇ СОЦІАЛЬНОЇ ВІДПОВІДАЛЬНОСТІ В РЕАЛІЯХ СЬОГОДЕННЯ Делотатчук А.В. ....	40
УПРАВЛІННЯ ФІНАНСОВОЮ СТРАТЕГІЄЮ ПІДПРИЄМСТВА У КРИЗОВИЙ ПЕРІОД Жила Н.О. ....	42
ФОРМУВАННЯ СТРАТЕГІЇ РОЗВИТКУ ПІДПРИЄМСТВА Пашіпровська Д.Ю. ....	44
<b>СЕКЦІЯ II. ДЕРЖАВНЕ УПРАВЛІННЯ ТА ЕКОЛОГІЯ</b>	
АКТУАЛЬНІ ПИТАННЯ РОЗВИТКУ ІНФРАСТРУКТУРИ СТАРОСТИНСЬКИХ ОКРУГІВ Мельник К.В. ....	48
ХІМІЧНА НЕБЕЗПЕКА Пеликаніч О.В. ....	48
<b>СЕКЦІЯ III. ТЕХНІЧНІ НАУКИ ТА ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ</b>	
ГОМОСТРУКТУРНІ ПТ ДЛЯ СЕНСОРНИХ МІКРОСИСТЕМ НА ФОНДІ ІНДІО Новосадий С.П., Бенько Т.Г. ....	52
ДОСЛІДЖЕННЯ НОСІВ ІНФОРМАЦІЇ З ЦИФРОВИХ ВІДЕОРЕЄСТРАТОРІВ ІНКВІЗІОН Копіца Я.Ю. ....	55
КОМПЛЕМЕНТАРНИЙ ФІЛЬТР ДЛЯ МІКРОМЕХАНІЧНОГО ІНКЛІНОМЕТРУ Строчак Г.Ю., Сперін О.М. ....	58
ПРО ОПТИМІЗАЦІЮ ТЕРМІНІВ ОНОВЛЕННЯ МОРСЬКОЇ ТРАНСПОРТНОЇ ІНФРАСТРУКТУРИ Малаксіано М.О. ....	61
РОЗРОБКА ІГРОВОГО ДОДАТКУ В ЖАНРІ RPG Сенишин С.М., Дібриний О.А. ....	64
СИСТЕМНИЙ АНАЛІЗ ПРОБЛЕМИ ОПТИМАЛЬНОГО СИНТЕЗУ ІНФОРМАЦІЙНО-ВИМІРЮВАЛЬНИХ СИСТЕМ Науково-дослідна група: Альошин Г.В., Коломійцев О.В., Третяк В.Ф., Сачук І.І., Петренко О.С. ....	65



# КОМПЛЕМЕНТАРНИЙ ФІЛЬТР ДЛЯ МІКРОМЕХАНІЧНОГО ІНКЛІНОМЕТРУ

Строгач Григорій Юрійович  
добробач вищої освіти приладобудівного факультету  
Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського», Україна

Савелін Олександр Миколайович  
асистент кафедри приладів і систем орієнтації і навігації  
Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського», Україна

Вступ. Потреба визначення кутового положення різних мобільних приладів, такі як смартфони, ігрові контролери, дрони, тощо унеможливила використання великогабаритних, важких і складних систем у їх складі. Рішенням становляться сучасні мікроелектромеханічні чутливі елементи систем орієнтації дозволяють широко використовувати класичні методи і підходи інерціальної навігації. Звичайно, мікроелектронні пристрої значно поступаються точністю класичним системам, але їх дешевизна та мініатюрність дозволяє використовувати їх у системах що не потребують високої точності [1].

Постановка задачі. Використання алгоритмів лише на основі акселерометрів приводить до хибності вихідних показань кутового положення тіла. Наприклад, продемонстрований у статті [2] інклінометр, задача якого вимірювати положення тіла відносно горизонту, має суттєвий недолік – неспроможність коректної роботи на рухомих основах. Приклад роботи інклінометру на рухомій основі представлено на (рис. 1).

На представлено (рис. 1) тестування роботи приладу проводилось на рухомих транспорті. Імітувались нахили, подібні до тих, які може здійснювати автомобіль на бездоріжжі, шляхом невеликих нахилів самого модулю. Вісь X була направлена по каліварно до напрямку руху транспорту, вісь Y – перпендикулярно до X. Як можна побачити з (рис. 1), вихідний кут осі X (червоний) досить не відображає значення нахилу. Через це становиться неможливим використання описаного у роботі [2] приладу на рухомих основах.

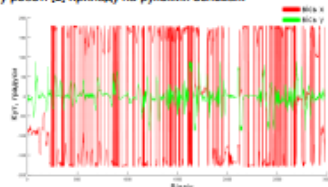


Рис. 1. Покази кутів на основі акселерометрів на рухомих основах

Для створення системи, що коректно працювала би на рухомій основі запропоновано використати підхід безлатентної інерціальної системи орієнтації,

використовуючи наробки [3], [4]. Однак, зважаючи на накопичення похибки від чисельного інтегрування виникає потреба компенсації відповірної похибки. Це можливо за допомогою різноманітних фільтрів, наприклад, фільтра на основі методу Калмана, комплементарного фільтра, фільтра Маджика, тощо. Найпростішим з перерахованих є комплементарний фільтр.

Принцип роботи комплементарного фільтра. Робота комплементарного фільтра полягає у комбінуванні вимірів акселерометрів та гіроскопів. Комбінування виконується за допомогою коефіцієнту фільтра, який визначає ступінь впливу показань блоку акселерометрів чи гіроскопів на вихідний кут. Наприклад за умови використання матриці напрямних косинусів (МНК) у якості кінематичного параметру орієнтації, визначення МНК для гіроскопів за допомогою рівняння Пуассона чисельно інтегруючи методом ліній прямокутника, для акселерометрів – складаючи МНК з кутів, що вираховуються відносно вектору сили вільного падіння, рівняння фільтра має вигляд [5], [6]:

$$C_{\text{out}}^{\text{out}} = (1-k) \cdot C_{\text{in}}^{\text{in}} + k \cdot C_{\text{in}}^{\text{a}}, \quad (1)$$

де  $k$  – коефіцієнт фільтра,  $C_{\text{in}}^{\text{in}}$  – МНК з показань гіроскопів,  $C_{\text{in}}^{\text{a}}$  – МНК з показань акселерометрів,  $C_{\text{out}}^{\text{out}}$  – вихідна МНК, з якої визначаються кути орієнтації.

Вибір величини коефіцієнту фільтра впливає на допоміжний емпіричний методу. Зважаючи на те, що розмірність коефіцієнту лежить в межах від 0 до 1 виберемо до розгляду такі значення: 0,9; 0,5; 0,1; 0,01. В якості сигналу використаємо значення з акселерометрів та гіроскопів записаних за допомогою мікроелектромеханічного датчика MPU6050. Для датчика використовувалась така серія дій навколо осі  $z$ : стан спокою, похил до 90 градусів в обидва боки навколо осі, стан спокою, нахил з середньою швидкістю, стан спокою, швидкий нахил стан спокою, імітування дії лінійних прискорень руху датчика у площині, стан спокою, випадковий нахил датчика стан спокою. Записаний сигнал зображено на (рис. 2).

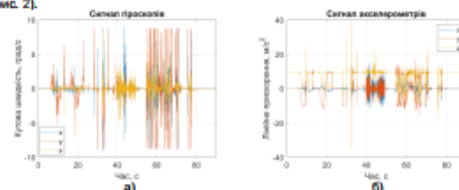


Рис. 2. Сигнали приладу: а) кутова швидкість; б) лінійне прискорення

Результати моделювання. Результат роботи фільтра з коефіцієнтами 0,9, 0,5, 0,1 та 0,01 зображено на (рис. 3). Як видно з графіків для коефіцієнтів 0,9 та 0,5 справедливе явище нестабільності вирахованих кутів та сильної невідповідності кутового положення при дії лінійних прискорень. Коефіцієнт 0,01 також не є сприятливим через сильну деградацію кутів та низьку швидкість.

Значення коефіцієнту 0,1 є найбільш сприятливим для використовуваного датчика. Зберігається швидкість відображення зміни положення системи, адекватне реагування на лінійне прискорення. За даного коефіцієнту швидкість дрейфу тангенса –  $4,185 \cdot 10^{-7}$  град/с, крену –  $1,0355 \cdot 10^{-7}$  град/с.

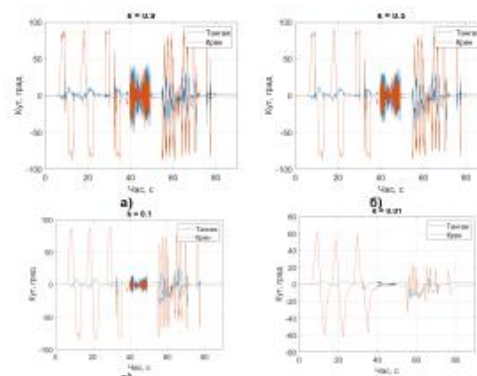


Рис. 3. Результат роботи фільтра для коефіцієнтів: а) 0,9; б) 0,5 в) 0,1; г) 0,01

Висновок. Комплементарний фільтр є гарним рішенням у задачах визначення кутового положення тіла відносно площини горизонту. Величина вибору коефіцієнту приладу впливає на його швидкість його роботи та точність значень вимірюваного куту. Занадто великий коефіцієнт буде допускати вплив акселерометрів у визначення кутів, занадто малий – негативно впливатиме на швидкість та коректність відображення кутів. Для кожного датчика коефіцієнт підбирається індивідуально та зважаючи на необхідну точність та швидкість системи, для якої необхідно визначити кутове положення.

## Список використаних джерел:

- Сисоєва, С. С. (2014). Тенденції ринку High-end MEMS-датчиків інерції. Нові уявлення: характеристики і застосування. *Комплекти та технології*, (5), 42–47.
- Строгач, Г. Ю., Савелін, О. М. (2019). Кутів на основі акселерометру ADXL-335. *Повідомлення приладобудівництва*, 45–48.
- Савелін, О. М., Бутайко, Д. В. (2016). Програмна модель системи орієнтації на базі MAX21105. *Діалог міжнародної науково-практичної конференції «ІНТ-2016»*, 55–58.
- Савелін, О. М., Наренко, М. В. (2017). Алгоритм безлатентної інерціальної системи орієнтації на лазерних гіроскопах. *Науковий вісник Національного технічного університету України «Київський політехнічний інститут»*, (3), 108–113.
- Tafiqul Islam, Ibt. Saifur Islam, Ibt. Shaikh-Ul-Islamud & Ibt. Hossain-E-Habib (2017). Comparison of complementary and Kalman filter based data fusion for attitude heading reference system. *AIP Conference Proceedings* 1919, 020002. Випущено з <https://doi.org/10.1063/1.5016520>
- Collier, S. (2007). The Kalman filter: a simple solution for integrating accelerometer and gyroscope measurements for a landing platform. *Chief Delphi white paper* (1) Випущено з [http://www.eec.sci.edu/techreports/techreports2\\_sakomplementary%20filter.pdf](http://www.eec.sci.edu/techreports/techreports2_sakomplementary%20filter.pdf)